UNIVERSITEIT
GENT

FACULTY OF **BIOSCIENCE ENGINEERING**

Department of Mathematical Modelling, Statistics and Bioinformatics

Meso-scale modeling of reaction-diffusion processes using cellular automata

ir. Pieter Van der Weeën

Thesis submitted in fulfillment of the requirements for the degree of
Doctor (Ph.D.) of Applied Biological Sciences

Academic year 2013-2014

Supervisor: Prof. dr. Bernard De Baets
Department of Mathematical Modelling,
Statistics and Bioinformatics
Ghent University, Belgium

Examination committee: Prof. dr. ir. Koen Dewettinck (Chairman)

Prof. dr. ir. Bruno De Meulenaer
Prof. dr. Samira El Yacoubi
Prof. dr. ir. Ingmar Nopens
dr. ir. Pieter Saveyn
Prof. dr. ir. Eveline I. P. Volcke

Dean: Prof. dr. ir. Guido Van Huylenbroeck

Rector: Prof. dr. Anne De Paepe

ir. Pieter Van der Weeën

MESO-SCALE MODELING OF REACTION-DIFFUSION PROCESSES USING CELLULAR AUTOMATA

Thesis submitted in fulfillment of the requirements for the degree of

Doctor (Ph.D) of Applied Biological Sciences

Academic year 2013-2014

Dutch translation of the title:
Mesoschaal modellering van reactie-diffusieprocessen aan de hand van cellulaire automaten

Please refer to this work as follows:
Pieter Van der Weeën (**2014**). *Meso-scale modeling of reaction-diffusion processes using cellular automata*, PhD Thesis, Department of Mathematical Modelling, Statistics and Bioinformatics, Ghent University, Ghent, Belgium.

ISBN 978-90-5989-720-5

Acknowledgments

It almost feels unreal that the time has come for the obligatory expressions of gratitude. I made a promise to myself that if this time ever came, I would go right to the point. So, here we go.

Along the way I had doubts about a lot of things. However, one thing that was always clear to me was that I had a promotor with an outstanding understanding of many different scientific fields and that had the exceptional gift of asking the right questions to guide me further along my path towards today. Therefore, I want to thank my promotor Bernard De Baets.

Vital during my journey were the many laughs and the cathartic talks with several colleagues. A special thanks to Arne, Elena, Elien, Hilde, Jan, Michiel, Séverine and Steffie.

Further, my parents deserve some words in this acknowledgment. It fills me with joy to know that you have always been proud of me and have always believed in me.

Finally, I have left the most important person till last (as one so often does). My gratitude goes out to my partner Tom for enduring the nervous breakdowns, for encouraging me to persist and for taking me as I am.

Thanks to all of you.

Pieter Van der Weeën
Gent 10/09/2014

Abbreviations

Abbreviation	**In full**
AE	Absolute error
AOP	Advanced oxidation process
CA	Cellular automaton
CB	Cocoa butter
CPES	Close-packing of equal spheres
CIP	Ciprofloxacin
DE	Differential equation
EE	Elementary Effects
FIM	Fisher information matrix
FQ	Fluoroquinolone
IR	Voltage drop
MOX	Moxifloxacin
PDE	Partial differential equation
RMSE	Root mean squared error
SA	Sensitivity analysis
SAE	Sum of absolute errors
SSE	Sum of squared errors
TAG	Triacylglycerol
TSM	Temporal series of micrographs
WE	Working electrode

Table of contents

1 General introduction

Mathematics has always interacted with other disciplines and has been influenced by them as well as provided to these disciplines a universal language to significantly advance their own fields of knowledge [1]. Examples of this interaction date back to the early days of human civilization. Summation series helped Ancient Egyptians build the pyramids and Ancient Greek philosophers studied the manner in which species evolve to fit their environment, which formed the basis for modern evolution theories. Since the beginning of the 20th century, mathematical models have become increasingly important in the life sciences. Examples of such models include the Michaelis-Menten equation for enzyme kinetics [2], Haldane's equation for genetic mapping [3], the Lotka-Volterra equations for predator-prey systems [4] and the Hodgkin-Huxley's equations for the neural axon membrane potential [5]. The long history between mathematical and life sciences proves that collaborations between scientists from different disciplines is a necessity and that they can lead to new methods, tools and approaches. Mathematical modeling of real-world problems attempts just that by bringing the different fields from which the problems arise in contact with mathematics and informatics.

Mathematical models, which are a translation of an artificial or natural system into a set of mathematical equations, are crucial in the life sciences and therefore their role has continued to increase. Indeed, these models can be used to support theories, however, they can also suggest the need for better experiments and more focused observations that in their turn can be employed to check the model accuracy. Further, large amounts of (experimental) data can be intelligently processed by o.a. efficient data mining algorithms and vizualisation tools. Ultimately, the integration of these large data sets should lead to models with predictive capabilities that, together with the ever increasing computational power, could partly replace the very costly and time consuming experimental analyses.

Different problems and processes require different models and methods and therefore, a variety of models have been developed. For example, ordinary, partial, integro-differential, functional and stochastic differential equations, methods of dynamical systems theory, neural networks, discrete mathematics approaches, tools from statistics and algebraic geometry, have all contributed to life sciences as such enriching the knowledge on them [1]. Nevertheless, compared to many physical and engineering systems, where mathematics has long played a key role in their understanding and quantification, the available knowledge of biological and (bio)chemical systems is still quite limited. Moreover, the complexity of biological systems, which arises from their heterogeneity, the large range of spatio-temporal scales and non-deterministic behavior, hampers the development of accurate mathematical models. For these reasons, this dissertation tries to contribute to a better understanding of a certain type of discrete mathematical models, namely cellular automaton based models, and the processes they model. The processes to be modeled are chosen from the broader field of chemical engineering and are all reaction-diffusion processes. The latter type of processes, which are naturally applied in chemistry, explains how the concentration of one or more substances distributed in space changes under the influence of two processes: local chemical reactions in which the substances are transformed into each other, and diffusion which causes the substances to spread out in space.

Part I gives some basic information about different subjects that are of importance in this dissertation. It starts by taking a closer look at mathematical modeling (Chapter 2) and goes deeper into one specific modeling paradigm, i.e. cellular automata (Chapter 3), which is the core modeling paradigm used throughout this dissertation. Further, Part I discusses the basics of sensitivity analysis, while elaborating on the Elementary Effects method (Chapter 4), since it is employed in the following parts of the dissertation.

Part II uses different case studies to examine the possible pitfalls and difficulties when modeling chemical systems using cellular automata. Every chapter in this part follows the same pattern: A problem is introduced, the acquisition of the experimental data is discussed, a mathematical model employing a cellular automaton is developed, the model is tested and validated and the conclusion of each chapter summarizes the key points of interest from each case study with regard to cellular automaton based modeling of chemical processes. The four different case studies that are discussed in Part II are reaction kinetics (Chapter 6), photocatalysis (Chapter 7), oil migration (Chapter 8) and pitting corrosion (Chapter 9).

From the different cases studies in Part II, it is clear that efficiently parameterizing the developed cellular automaton based models is tedious. Therefore, in Part III an improved grid search method based on the information gained from a global sensitivity analysis is introduced. Firstly, the theory behind the proposed approach is discussed in detail (Chapter 11) after which the different introduced concepts are illustrated making use of a simple linear model and two well-known nonlinear models (Chapter 12). Further in Chapter 12, it is verified whether or not the proposed approach gives rise to better results in practice by revisiting the parameterization of one of the case study models from Part II by the strategy described in this part.

In the final part, i.e. Part IV, some general conclusions are drawn. These conclusions are written in the form of a checklist for good practice when using CA-based models in (bio)chemical engineering, based on the information gathered through a literature survey and the case studies in Part II. Further, a Dutch summary of this disseration can be found in this part.

PART I

PRELIMINARIES

2 Mathematical modeling

Mathematical models describe how people believe the world functions. In mathematical modeling, the process of developing a mathematical model, those beliefs are translated into mathematical concepts and language (see Figure 2.1). Mathematical models are used not only in the natural sciences (e.g. physics, biology, earth science and meteorology) and engineering disciplines (e.g. computer science and artificial intelligence), but also in the social sciences (e.g. economics, psychology, sociology and political science). An increasing number of researchers and companies, partly due to the ever growing computational power and speed and the overall knowledge on mathematical modeling, are employing mathematical models in their research or company.

There exist many introductory text books on mathematical modeling. All concepts that are introduced and discussed in this chapter, and much more, are delt with at length in e.g. Aris [6], Bender [7], Gershenfeld [8] and Shearer *et al.* [9].

2.1. Building blocks

A mathematical model usually describes a system by a set of variables and a set of equations that establishes relationships between the variables. Relationships can be described by operators, such as algebraic operators, functions, differential operators, ... Variables represent some properties of

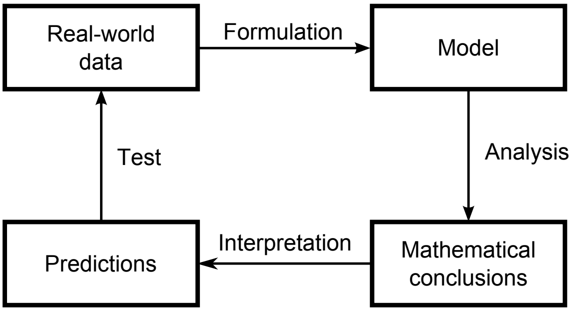


Figure 2.1: Schematic representation of the process of mathematical modeling.

the systems that are of interest and that can be quantified. Different types of variables can be discerned. Dependent variables are model constituents of which the value changes in function of the independent variables (such as time and space). Inputs, outputs and states are therefore variables and are named input variables, output variables and state variables, respectively. Finally, there exist also exogenous variables which are sometimes known as constants or parameters. The former have the same value, regardless of the application, while the latter have a value that depends on the application.

Objectives and constraints of the system and its users can be represented as functions of the output variables or state variables. Although there is no limit to the number of objective functions and constraints a model can have, using or optimizing the model becomes more involved (computationally) as this number increases.

2.2. Classification

There are many different ways to classify models. In what follows, some of the more popular ways are discussed.

Linear vs. nonlinear One possible way to distinguish between linear and nonlinear models is the following: If all the operators in a mathematical model exhibit linearity, the resulting mathematical model is defined as linear. A model is considered to be nonlinear otherwise. However, the definition of linearity and nonlinearity is dependent on the context and linear models may have nonlinear expressions in them. Nonlinearity, even in fairly simple systems, is often associated with phenomena such as chaos and irreversibility. Although there are exceptions, nonlinear systems and models tend to be more difficult to study than linear ones. A common approach to nonlinear problems is linearization, but this can be problematic if one is trying to study aspects such as irreversibility, which are strongly tied to nonlinearity.

Static vs. dynamic A dynamic model accounts for time-dependent changes in the state of the system, while a static (or steady-state) model calculates the system in equilibrium, and thus is time-invariant. Dynamic models typically are represented by differential equations.

Explicit vs. implicit If all of the input variables of the overall model are known, and the output variables can be calculated by a finite sequence of computations, the model is said to be explicit. But sometimes it are the output parameters that are known, and the corresponding inputs are solved by an iterative procedure, in which case the model is termed implicit.

Discrete vs. continuous A discrete model treats objects as discrete, such as the particles in a molecular model or the states in a statistical model. A continuous model represents the objects in a continuous manner, such as the velocity field of fluid in pipe flows and temperatures and stresses in a solid.

Deterministic vs. stochastic A deterministic model is one in which every set of variable states is uniquely determined by parameters in the model and by sets of previous states of these variables. Therefore, deterministic models perform the same way for a given set of initial conditions. Conversely, in a stochastic model, randomness is present and variable states are not described by unique values, but rather by probability distributions.

Black-box vs. white-box A black-box model simply reproduces the input-output behavior of the system, but makes no use of *a priori* information on the process at hand [10]. White-box models on the other hand are based on a detailed understanding of the underlying physical laws and processes. Usually it is preferable to use as much *a priori* information as possible to make the model more accurate. Therefore the white-box models are usually considered easier, because if the information is handled correctly, the model will behave correctly. Often the *a priori* information comes in the form of knowing the type of functions relating different variables. Sometimes it is useful to incorporate subjective information into a mathematical model. This can be done based on intuition, experience, or expert opinion, or based on convenience of mathematical form.

Nevertheless, practically all systems are gray-box models, meaning that they are somewhere between the black-box and white-box models. Like the black-box models, the gray-box models are parametric models where the parameters are learned from data, but the difference with black-box models is that here the parametric model is motivated by basic physical ideas instead of simply being selected from a class of universal approximators. Usually gray-box models, in comparison to black-box models, are characterized by smaller parameter sets and the parameters themselves yield some physical insight into the problem.

2.3. Advantages and limitations

The use of mathematical models entails a number of advantages. First of all, mathematics is a very concise and precise language, which facilitates the formulation of ideas and identification of the underlying assumptions. Further, all the results that mathematicians have proven over hundreds of years are available, resulting in a well-defined modeling toolbox.

When studying new processes or processes that are poorly understood, a lot of knowledge is gathered via experimental analysis in the laboratorium. However, this analysis is time consuming and expensive, making mathematical modeling an attractive alternative since it offers a large gain in time, because computers can be used to perform numerical calculations. Still, the model first has to be developed and implemented, which costs some time as well. Furthermore, experimental analysis is not redundant when performing virtual experiments: Sound data is still essential for model calibration and validation.

In the case of an existing, well-known system, modeling allows for scenario analysis. The latter means that many different scenarios can be computed in a relatively short time. This allows for example to estimate the effect of different chemical production processes on humans and the environment or the yield of the chemical production process under various operating conditions. Another use of modeling in this case is process optimization, where modeling facilitates the search for an optimal design. Finally, operational optimization and control are also mentioned here as important modeling incentives.

The main limitation of using mathematical models is that any model is not the real system, it is always an approximation. Therefore, there is never a perfect model and results can be influenced by model inaccuracies. There is a large element of compromise in mathematical modeling. The majority of interacting systems in the real world are far too complicated to model in their entirety (cf. Occam's razor [11]). Hence, the most important parts of the system need to be identified and included in the model, while the rest is excluded. For that reason, the task of the modeler consists of selecting the appropriate parts and ensuring that a good agreement with the experimental measurements can be achieved with the model.

2.4. Spatio-temporal modeling

Spatial-temporal models arise when data is collected across space as well as time. Until recently, there has not been a theory of spatial-temporal processes separate from the already well-established theories of spatial statistics and time series analysis [12]. For example, the methods for the fitting of spatio-temporal models are mostly the same as for time series and spatial models, i.e. least squares methods, maximum likelihood estimation or Bayesian inference, but there is a lack of general purpose software when it comes to fitting these models [13, 14]. However, due to the proliferation of data sets that are both spatially and temporally indexed, spatial-temporal modeling has received dramatically increased attention in the last few years.

Within the field of (bio)chemistry, spatio-temporal processes, and therefore also spatio-temporal models, constitute an important category. The modeling of (bio)chemical processes up till now has been performed in various ways, some of the more popular ones being empirical approaches, stochastic models and (partial) differential equations ((P)DEs). Many of these models show a good performance and are successful at describing natural phenomena in a mathematical way [15]. Catalyzed by the development of extremely powerful, high-speed computers, two novel approaches to study (bio)chemistry emerged in recent years, being molecular dynamics and Monte Carlo calculations [16, 17]. These models rely on the interaction of atoms and molecules in a system via electric and steric forces, expressed in terms of force fields. Both of them have great strengths and often lead to similar results for the properties of the system under investigation. On the other hand, these methods depend on elaborate models for the molecular interactions, which results in computationally demanding calculations.

The most popular and also classical way to model spatio-temporal chemical processes is by means of (P)DEs [18, 19]. These established mathematical constructs, of which the state, space and time domains are continuous, play an important role in modeling virtually every physical, chemical, technical, or biological process, from celestial motion, to bridge design, and interactions between neurons. The (bio)chemical process to be modeled is broken down to its fundamental building blocks and for each of these blocks a (P)DE describes the underlying physical phenomenon, resulting in a rigorous mathematical description of the macroscopic process at hand. Different aspects of (P)DEs, mostly concerned with their solutions, have been and are being studied, which has led to the availability of a myriad of mathematical tools to study (P)DE-based models. Furthermore, these models are often transparant in terms of how a change in parameter values affects the outcome and are therefore relatively interpretable [20].

Nevertheless, there are some downsides to the use of (P)DEs. Finding and implementing a system of (P)DEs for a specific process still remains tedious and when such a system of PDEs is used to describe real-world problems, it may not necessarily be directly solvable, i.e. it does not have closed-form solutions. In this case, solutions have to be approximated using numerical methods that unavoidably give rise to approximation errors and often also to stability problems [21, 22]. Furthermore, spatial homogeneity is frequently assumed when (P)DEs are employed to describe the system's dynamics even when this is a strong simplification of reality. Therefore, researchers often look for novel modeling methods to better suit their needs.

Of course, (P)DEs are not the only mathematical formalism to describe spatio-temporal processes. Table 2.1 shows eight different types of spatio-temporal modeling paradigms depending on the state, space and time domain being continuous or discrete [23]. This dissertation will focus on the use of cellular automata (CAs) for the modeling of spatio-temporal processes within the broader field of chemistry and the possibilities and (practical) limitations that go together with this.

Table 2.1: Spatio-temporal modeling paradigms [23].

State	Space	Time	Name
continuous	discrete	discrete	coupled map lattice
continuous	discrete	continuous	reaction-dispersal network
continuous	continuous	discrete	reaction-dispersal network
continuous	continuous	continuous	partial differential equation
discrete	continuous	discrete	neighborhood model
discrete	continuous	continuous	spatial point process
discrete	discrete	discrete	cellular automaton
discrete	discrete	continuous	interacting particle system

3 Cellular automata

In this chapter, a short introduction to CAs is given, followed by the formal, general mathematical definition of a CA, which will be adhered to throughout this dissertation. Some parts of this definition are elaborated on with special attention for the different updating mechanisms. For the case studies of Part II, the different parts of this definition will be further specified. Finally, advantages and disadvantages of the use of CAs are discussed.

3.1. Introduction

A CA is a collection of cells on a tessellation, i.e. they are spatially discrete, where each cell changes its discrete state as a function of time according to a given transition function, based on the cell's own state and the states of its neighboring cells. This transition function is applied iteratively for as many discrete time steps as desired. CAs are discrete, abstract computational systems that have proven useful both as general models of complexity and as more specific representations of non-linear dynamics in a variety of scientific fields. CAs appeared for the first time in literature in the first half of the 20th century under the name of 'cellular spaces'. This early work forms the starting point for research in this field, although the real breakthrough of CAs in the academic world began with the publication of Conway's Game of Life in the early 70s [24] and Wolfram's pioneering work [25, 26].

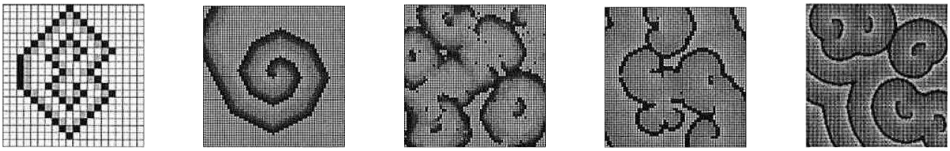
Today, CAs have been successfully employed in the study of several biological and abiological processes, overcoming some of the issues addressed in the previous chapter inherent to the more classical modeling methods as well as providing researchers with novel insight in the processes. The domains where CA-based models have been proposed range from traffic control [27] over economics [28] and politics [29] to applied biological sciences. One of the first attempts to model chemical processes by means of discrete models was the HPP model (after the inventors Hardy, Pomeau and de Pazzis) [30, 31], which was in fact the first Lattice Boltzmann model. The latter model makes use of the discrete Boltzmann equation, instead of the

Navier-Stokes equations, to simulate the flow of a Newtonian fluid, where the particles move alongside the edges of the tessellation and do not occupy the cells. In the HPP model, the tessellation is square, and particles can move to any of the four sites whose cells share a common edge. If two particles collide head-on, the outcome will be two particles leaving the site at right angles to the direction they came in. This model lacked rotational invariance, which made it highly anisotropic. To overcome this problem, the FHP model (after the inventors Frisch, Hasslacher and Pomeau) was developed on a hexagonal grid and with particles with six or seven different velocities [30, 31].

Through the years, CA-based models have been increasingly used to model different types of reaction-diffusion processes. The stochastic CA-based model to describe the diffusion equation as proposed by Chopard and Droz [32, 33] is able to simulate well the simultaneous random walk of many particles. The parameters of the CA's transition function allow for an adjustment of the diffusion constant and play an important role in the isotropy of the dynamics, while making the model suitable to describe a range of reaction-diffusion processes [33]. Further, Kier *et al.* developed a CA-based model to describe the interaction between two H_2O molecules employing a stochastic CA [15]. Their CA-based model is used to model, a.o., the solid-liquid interface [34], the bulk structure of water [35] and solute diffusion through water [36].

A remarkable type of reaction-diffusion process is the oscillating reaction, where the concentration of one or more components undergoes periodic changes in time and/or space [37]. Normally, chemical reactions are characterized by a continuous decrease of substrate and a corresponding continuous increase of reaction products. However, for oscillating reactions, the change in concentration of intermediate products is not continuous and this undulating behavior results in specific spatio-temporal patterns. Figure 3.1 shows some of these patterns for the Belousov-Zhabotinsky reaction [38] taking place on a petri dish. CAs, such as the one used in the Greenberg-Hasting model [39], are capable of reproducing the spatial patterns of oscillating reactions thanks to their discrete tessellation. Figures 3.2(a)–3.2(e) show simulated outcomes at given moments in time using the Greenberg-Hasting model with different parameter settings [40]. The spiral-shaped patterns can be clearly observed where the black line represents the excitation front. This excitation front is formed by excited cells

Figure 3.1: Examples of spiral-shaped patterns formed by the Belousov-Zhabotinsky reaction on a petri dish [38].



(a) (b) (c) (d) (e)

Figure 3.2: Simulated outcomes at given moments in time using the Greenberg-Hasting model with different parameter settings [40].

and can itself excite new cells. The disadvantage of using CAs to model oscillating reactions is that usually a smooth curvature is not easily attainable (see Figures 3.2(a) and 3.2(b)) unless large neighborhoods are considered, which results in extensive calculation times.

Of course, reaction-diffusion processes are not the only type of process modeled by means of CAs over the years. Other authors cover, a.o., the use of CA-based models in chemistry to model a recrystallization process [41] or to study the structural chemistry of molecules [42]. Furthermore, the last years, CA-based models have also been introduced in the broader field of chemical engineering and are being used to tackle problems in the domains of biochemistry [43, 44, 45, 46], biotechnology [47, 48] and food chemistry [49]. Nevertheless, despite the growing interest in CA-based models, publications on the use of such models within the field of (bio)chemical engineering are still scarce, due to the limitations inherent to CAs (see infra).

It is therefore the goal of this dissertation to investigate the potential of CA-based modeling for spatio-temporal (bio)chemical processes. On the one hand, these models could provide researchers with a wider range of modeling tools and, in some complex cases, a solution to problems encountered with some of the more classical modeling methods. Furthermore, it would allow to make better use of the increasingly more detailed spatio-temporal data that is available from e.g. microscopy and tomography. On the other hand, most researchers experience a number of practical difficulties when working with CA-based models in the broader field of chemical engineering. In addition, many reports on modeling efforts in this field are incomplete, therefore making it hard to learn from each other. This dissertation explores the various difficulties modelers encounter with CA-based modeling through the study of four case studies in Part II. Each of these case studies will highlight different modeling choices or possibilities, as such providing a variety of information. Besides, more than just summing up the potential pitfalls, this dissertation tries to provide the means to deal with them. Nevertheless, the conclusions drawn throughout this dissertation are based on these four case studies, which are selected for their increasing complexity of the underlying phenomenon and the availability of experimental data, and are therefore just a first step in the development of a general framework that is valid for a broad class of spatio-temporal (bio)chemical processes.

3.2. Paradigm

In this dissertation, the following definition of a CA is relied upon.

Definition 1 *(Cellular automaton)*
A cellular automaton $\mathcal{C}$ can be represented as a quintuple

$$\mathcal{C} = \langle \mathcal{T}, S, s, N, \Phi \rangle \ ,$$

where

(i) $\mathcal{T}$ is a countably infinite tessellation of a d-dimensional Euclidean space $\mathbb{R}^d$, consisting of cells c_i, $i \in \mathbb{N}$.

(ii) S is a finite set of h states, often $S \subset \mathbb{N}$.

(iii) The output function $s : \mathcal{T} \times \mathbb{N} \to S$ yields the state value of cell c_i at the t-th discrete time step, i.e. $s(c_i, t)$.

(iv) The neighborhood function $N : \mathcal{T} \to \bigcup_{q=1}^{\infty} \mathcal{T}^q$ maps every cell c_i to a

finite sequence $N(c_i) = \left(c_{i_j}\right)_{j=1}^{|N(c_i)|}$, *consisting of* $|N(c_i)|$ *distinct cells* c_{i_j}.

(v) $\Phi = (\phi_i)_{i\in\mathbb{N}}$ *is a family of functions*

$$\phi_i : S^{|N(c_i)|} \to S\,,$$

with each ϕ_i *governing the dynamics of cell* c_i*, i.e.*

$$s(c_i, t+1) = \phi_i\Big(\tilde{s}(N(c_i), t)\Big),$$

where $\tilde{s}(N(c_i), t) = \left(s(c_{i_j}, t)\right)_{j=1}^{|N(c_i)|}$.

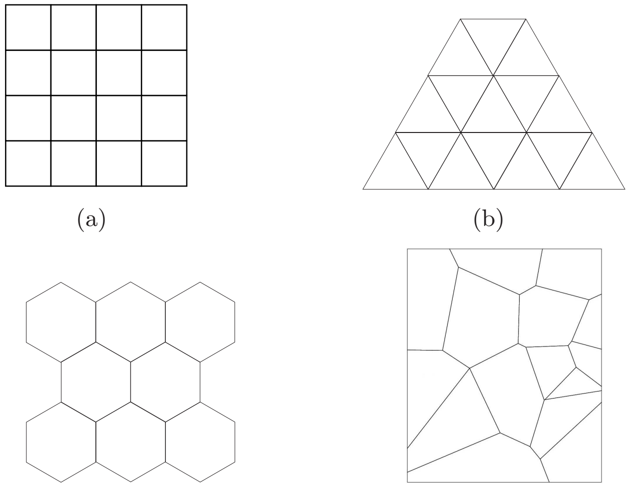
For reasons of clarity, the different parts of this definition will be elaborated on in the remainder of this section.

3.2.1. Tessellation

A tessellation $\mathcal{T}$ is a collection of spatial entities that fills the Euclidean space $\mathbb{R}^d$ without overlap or gaps, with mostly $d \in \{1, 2, 3\}$. A regular 2D (resp. 3D) tessellation is a highly symmetric tessellation made up of congruent regular polygons (resp. polyhedrons), where consequently every cell c_i of $\mathcal{T}$ has the same number of neighbors. The three most popular regular 2D tessellations are those which consist of squares (see Figure 3.3(a)), triangles (see Figure 3.3(b)) or hexagons (see Figure 3.3(c)). Often a tessellation of $\mathbb{R}^2$ that consists of squares is used in literature [50, 51, 52], because it has the most straightforward implementation and suffices for most modeling purposes.

Although these regular tessellations are widely used, some phenomena cannot be explained by employing CAs with these regular tessellations. Points of concern are, a.o., the fixed number of neighbors and the lack of isotropy. To overcome the former problem, CAs on irregular tessellations were developed (see Figure 3.3(d)) [53]. Such tessellations can be constructed in an elegant and straightforward way, but are less efficient to perform calculations with.

Further, the infinite nature of $\mathcal{T}$ hampers the practical use of CAs, since for computer simulations, only a finite number of cells can be considered. Furthermore, the applications of CAs in (bio)chemistry are not taking place



(a) (b) (c) (d)

Figure 3.3: Different tessellations of $\mathbb{R}^2$ composed of (a) squares, (b) triangles, (c) hexagons and (d) irregular cells.

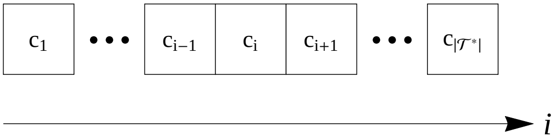


Figure 3.4: Indexing of the cells of a 1D CA.

in infinite space. Therefore, a finite tessellation $\mathcal{T}^*$, consisting out of $|\mathcal{T}^*|$ cells is normally used. An indexing of the cells of a regular 1D, 2D and 3D CA is introduced in Figures 3.4, 3.5 and 3.6, respectively.

Finally, the importance of the tessellation size $|\mathcal{T}^*|$ on the model calculation times is mentioned here. On the one hand, a low number of cells implies a low number of evaluations and therefore faster calculation than when a large number of cells is employed. On the other hand, a sufficiently high number of cells is needed to get accurate simulation results. Consequently, the tessellation size $|\mathcal{T}^*|$ has to be as small as possible without compromising the accuracy the results produced.

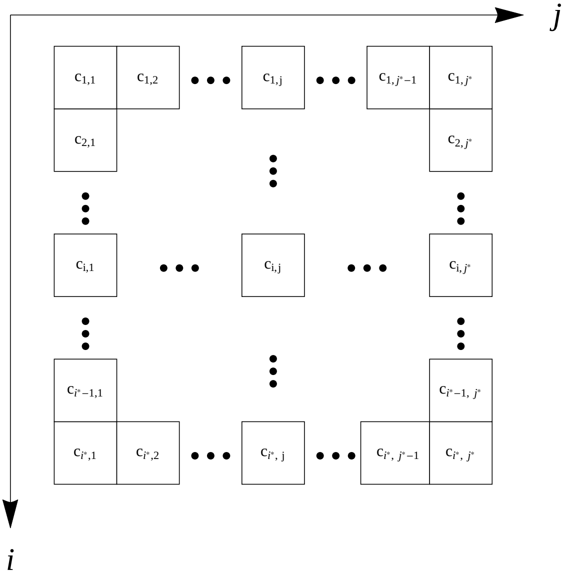


Figure 3.5: Indexing of the cells of a 2D CA.

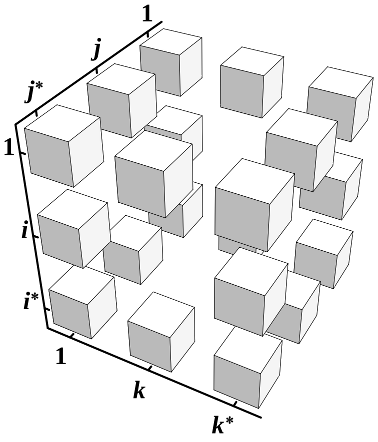


Figure 3.6: Indexing of the cells of a 3D CA.

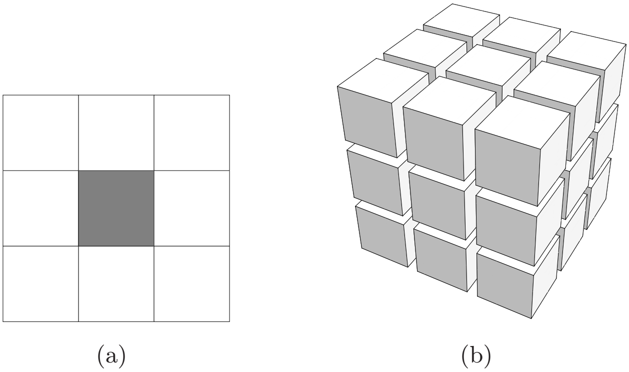

(a) (b)

Figure 3.7: (a) 2D and (b) 3D Moore neighborhood of a cell c_i (gray) in a (a) square and (b) cubic tessellation.

3.2.2. Discrete states

Every cell c_i takes as value one of the h discrete values comprised in the set S. The state of the cells of $\mathcal{T}^*$ at $t = 0$, i.e. $s(c_i, 0)$, forms the initial condition of $\mathcal{T}^*$. Spatially seen, an even distribution of the different discrete states is, technically, the most straightforward initial condition, which explains its popularity as initial condition in literature. However, this initial condition is not easily obtained in the laboratory [23].

3.2.3. Neighborhood function

Many different neighborhood functions can be defined, the two most important ones on regular tessellations being the Moore and the von Neumann neighborhood. The Moore neighborhood of c_i comprises those cells that share at least a vertex with cell c_i. The von Neumann neighborhood is a more restricted neighborhood in which only those cells that share an edge with the central cell c_i are considered as neighbors. Figures 3.7(a) and 3.7(b) show the nine and 27 neighbors in the 2D and 3D radius one Moore neighborhood, respectively, while Figures 3.8(a) and 3.8(b) show the five and seven neighbors in the 2D and 3D radius one von Neumann neighborhood, respectively.

The radius one neighborhoods are the most occuring, although in this dissertation, also the radius two 2D Moore neighborhood is used. Figure 3.9 shows the 2D radius two Moore neighborhood on a square tessella-

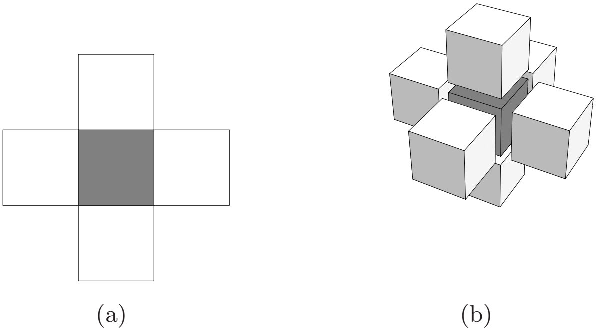


Figure 3.8: (a) 2D and (b) 3D von Neumann neighborhood of a cell c_i (gray) in a (a) square and (b) cubic tessellation.

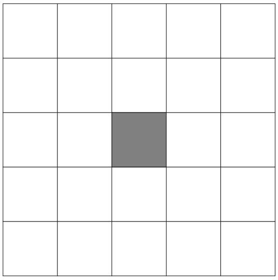


Figure 3.9: 2D radius two Moore neighborhood of a cell c_i (gray) on a square tessellation.

tion.

The finite nature of $\mathcal{T}^*$ calls for boundary conditions. In this case, the boundary cells of the tessellation are possibly, depending on the type of boundary condition, governed by a different transition function than the rest of the cells. The three most common boundary conditions are fixed, periodic and absorbing boundaries. Fixed boundaries act like a wall through which no movement or contact is possible. For instance, Figure 3.10(a) shows a 5×5 square tessellation in which the transition function states that a cell in state 1, i.e. a black cell, takes the state of the cell left of it and that a cell in state 0, i.e. a white cell, takes the state of the cell right of it. The resulting tessellation at $t = 1$ is depicted in Figure 3.10(b). Figure 3.10(c) shows the result at $t = 2$ when fixed boundaries apply. The transition function for boundary cells when fixed boundaries apply states that the cell has no neighbors past the borders, such that the cell in state

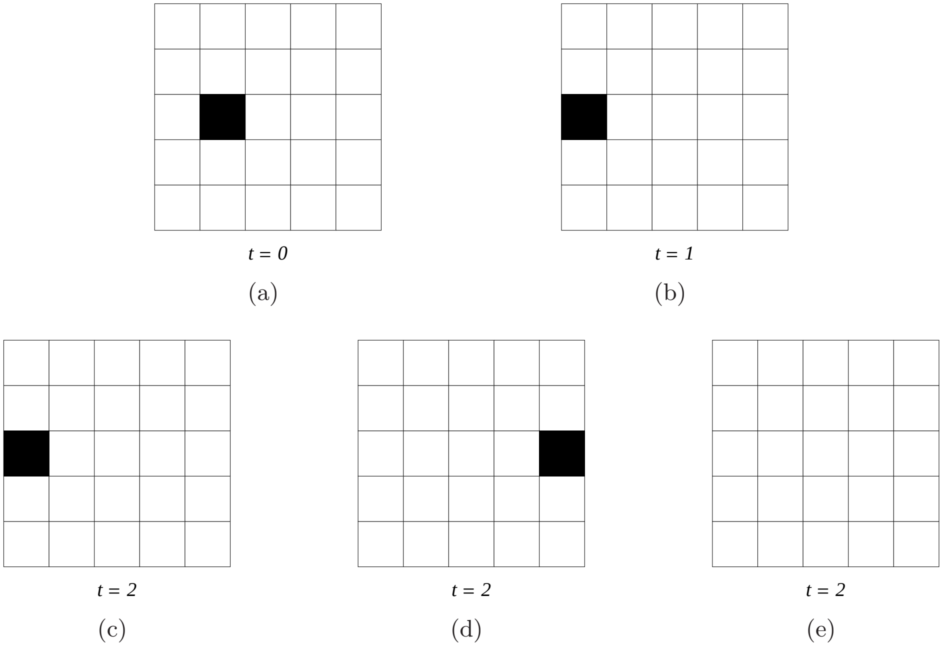


Figure 3.10: Square tessellation in which the transition function states that a cell in state 1 (black) takes the state of the cell left of it and that a cell in state 0 (white) takes the state of the cell right of it at (a) time step $t = 0$, (b) time step $t = 1$ and time step $t = 2$ when (c) fixed boundaries, (d) periodic boundaries and (e) absorbing boundaries apply.

1 remains unchanged.

The second type of boundary condition are the periodic boundaries where the opposite border along the same dimension is considered the neighboring border, so that in practice no boundary cells exist in this case. For instance, for a regular 2D CA the latter implies that the cells of the first row (column) are considered the neighbors of the cells of the last row (column) (see Figure 3.10(d)). Periodic boundaries are the most popular ones since N is the same for all cells and furthermore, they simulate the infinite nature that was lost by going from $\mathcal{T}$ to $\mathcal{T}^*$. It should be mentioned that the former only holds for a regular tessellation, since not necessarily every cell of an irregular tessellation has the same number of neighbors. Finally, the resulting tessellation at $t = 2$ when absorbing boundaries apply is shown in Figure 3.10(e). Here, the transition function of the boundary cells states that all boundary cells in a state different from 0 are set to 0, as such 'absorbing' the states.

3.2.4. Transition function

The transition function Φ determines the state of a cell c_i at the $(t+1)$-th time step based on the cell's current state and the states of its neighboring cells, $\tilde{s}(N(c_i), t)$. Two important CA families can be distinguished, based on the way the transition function is applied [15]. A first one is the family of deterministic CA-based models, which is the classical way of applying Φ and is also used in Definition 1. Here, the transition function is always applied, i.e. with probability P equal to one. A second family is that of stochastic CA-based models, which is used regularly when (bio)chemical processes are concerned. In this case, the transition function Φ is subject to a probability P, determining whether or not the transition function is executed (see Definition 2). This entails that the outcome of a stochastic CA-based model differs for every simulation.

Definition 2 *(Stochastic cellular automaton)*
A stochastic cellular automaton is a CA fulfilling premises $(i)-(iv)$ *of Definition 1, and for which there exists a* $P \in]0,1[$ *such that*

$$s(c_i, t+1) = \begin{cases} \phi_i\Big(\tilde{s}(N(c_i), t)\Big) & \text{, with probability } P, \\ s(c_i, t) & \text{, with probability } 1-P. \end{cases}$$

Further, it can be noticed that not necessarily every cell c_i has to possess its own transition function ϕ_i. In the case of homogeneous CA there is a single transition function Ψ that governs the dynamics of all cells. Homogeneous CAs can be defined according to Definition 3.

Definition 3 *(Homogeneous cellular automaton)*
A homogeneous cellular automaton is a CA fulfilling premises $(i)-(iv)$ *of Definition 1, and for which there exists a* $\Psi : \bigcup_{h\in\mathbb{N}} \mathcal{M}_h(S) \to S$ *such that*

$$s(c_i, t+1) = \Psi\Big(\tilde{s}(N(c_i), t)\Big),$$

where $\mathcal{M}_h$ *denotes the set of all multisets of cardinality* h *in* S.

In this dissertation, only homogeneous CAs are discussed, with exception for the transition function of the boundary cells, depending on the applied boundary conditions (see supra). Besides the distinction between updating the CA in a deterministic or probabilistic manner, it should be mentioned that Ψ can be applied either synchronously, i.e. Ψ is applied at each time step for every cell $c_i \in \mathcal{T}^*$, or asynchronously, whereby cells are not updated simultaneously. Because this is such an important issue when modeling (bio)chemical processes, as will be shown in this dissertation, the next section deals with the different updating methods.

3.3. Synchronous and asynchronous updating

3.3.1. Choice of update method

When performing simulations with CAs, a decision has to be made on how to apply the transition function Ψ. This can be done either synchronously or asynchronously. A synchronous update of the cells c_i of $\mathcal{T}^*$ means that all cells are updated simultaneously according to Ψ at every consecutive time step. Although this approach is widely used and effective in many cases, questions arise when synchronous CAs are used to model physical phenomena, since nature does not have a universal clock and, hence, the synchronous update method is rather artificial [38]. To overcome this drawback, asynchronous update methods have been developed [54, 55, 56, 57, 58].

Updating asynchronously can result in qualitatively and quantitatively different dynamical behavior of CAs as compared to updating synchronously [38, 54], especially when stochastic components are involved [54]. For example, it has been observed that cyclic patterns as well as various other patterns can only occur with synchronous updating. Further, the stationary states that can be reached with a CA are identical for both synchronous and asynchronous updating. However, which stationary state is reached starting from a given initial state may depend on the update method used. Moreover, the influence of the different update methods depends strongly on the actual transition function employed.

Most researchers opt for synchronous updating in their work [43, 49, 59], as it complies with the CA paradigm as conceived originally and it also entails a more straightforward implementation compared to asynchronous update methods. However, the choice for a given update method should be motivated by the process one wants to study: if a discrete time step of the CA corresponds to a relatively large time interval, expressed in physical time units, it may be assumed that every cell in $\mathcal{T}^*$ will be updated during each time step and synchronism is justified [54]. Yet, as the physical time that is covered per time step becomes smaller, the number of events that occur during its elapse decreases and asynchronism becomes a more natural option.

Further, asynchronous updating can help to respect the law of mass conservation, which is of interest when modeling (bio)chemical processes. Consider, for example, the diffusion process, which is often simulated by switching a cell's state with the state of the cell it diffuses towards. When at a discrete time step n cells try to switch state with the same cell, this leads in a synchronous update to the reduction of the total number of diffusing states by $n - 1$. Moreover, this reduction can take place at different places of the tessellation and at every time step, potentially resulting in an unacceptable loss in mass. Asynchronous update methods circumvent this problem by evaluating cell by cell and respect as such the law of mass conservation.

In the remainder of this section different synchronous and asynchronous update methods are discussed [38, 54, 60]. In order to discriminate easily among the different methods, the following parameters are introduced [60]:

(i) E_i represents the number of time steps between two consecutive updates of a cell c_i, i.e. how many time steps it takes for c_i to be updated.

(ii) F_i represents the delay of update, i.e. the number of time steps it takes before cell c_i is updated for the first time.

(iii) G_t represents the number of cells c_i that are being updated at time step t.

3.3.2. Synchronous update methods

Every cell c_i in $\mathcal{T}^*$ is updated in parallel at each time step, when synchronous updating is concerned. This update method is represented in Figure 3.11 where the gray cells indicate which cells are updated at a specific time step. As can be seen, all cells are updated simultaneously at every time step, such that it may be written

$$\forall\ i\ :\ E_i = 1,\ F_i = 0\,,$$

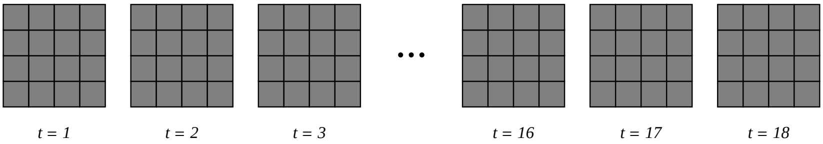


Figure 3.11: Representation of the cells being updated (gray) at time steps 1, 2, 3, 16, 17 and 18 according to the synchronous update method.

$$\forall\ t\ :\ G_t = |\mathcal{T}^*|\ .$$

Instead of choosing an asynchronous update method to comply with the law of mass conservation, one can also opt for the adjustment of the synchronous update method to obey the law of mass conservation. In some cases, it is also possible to employ a block CA, which is a special type of synchronous updating. Both are discussed next.

Adjustment to obey the law of mass conservation The most common way to do so is by looking at the intention of the transition function rather than directly executing it [32, 33, 61, 62]. For the case of diffusion, this comes down to constructing both a list of the locations of all cells that are allowed to switch their state in that specific time step and a list of the locations of the cells they want to switch state with. Next, a list of conflicts is constructed containing the cells that try to switch state with the same cell, as well as those cells that try to switch state with a cell that at that time step also wants to switch state with another cell. The cells on this conflict list are not allowed to switch state in that particular time step, as such getting around the issue of mass disappearance.

Although a valid and often used technique, it does nullify to some extent the implementation convenience and time and memory gain synchronous updating has over asynchronous updating. Furthermore, when a real-world interpretation of the model parameters through parameterization is desired, large conflict lists, and therefore many incomplete evaluations, influence the obtained parameter values and their interpretability.

Block cellular automaton Block CAs are commonly used to implement lattice gases, a type of CA used to simulate fluid flows, and other quasi-physical simulations, since it is straightforward to construct a transition function that obeys physical constraints such as reversibility and conservation laws [26, 63]. Firstly, the tessellation is split up into non-overlapping blocks, where each block has the same size and shape. The simplest partitioning of the tessellation is the Margolus neighborhood, where the tessellation is divided into 2-cell blocks (or 2×2 squares in two dimensions, or $2 \times 2 \times 2$ cubes in three dimensions, etc.) [63].

In each time step, the transition function is applied to a whole block at a time rather than a single cell and all blocks are evaluated simultaneously, as such making it a special case of synchronous updating. It is important to note that during the application of the transition function, no communication is possible between the different blocks and that therefore all processes take place between the molecules present in the same block. Finally, after the blocks are evaluated, the tessellation is reassembled and a rule for shifting the partition is applied. The latter is necessary to ensure that the pattern of cell states changes over time to perform some nontrivial computation or simulation. The shifting rule usually accompanying the Margolus neighborhood dictates that every block of 2 cells is shifted by one cell (along each dimension) in alternate time steps [63]. The use of a block CA speeds up calculations, facilitates implementation, obeys mass conservation laws and eliminates the need for boundary conditions. However, it can only be employed for well-mixed systems where the physical boundaries are not included in the tessellation.

3.3.3. Asynchronous update methods

There are different asynchronous update methods described in literature. Here, four of the most popular ones are elaborated on.

Random independent update method At each time step a single cell c_i is selected at random from $\mathcal{T}^*$ (see Figure 3.12). For this update method, it may be written

$$\forall\ i\ :\ E_i \sim \mathcal{H}\left(\frac{1}{|\mathcal{T}^*|}\right),\ F_i \sim \mathcal{H}\left(\frac{1}{|\mathcal{T}^*|}\right),$$

$$\forall\ t\ :\ G_t = 1\,,$$

where $\mathcal{H}\left(\frac{1}{|\mathcal{T}^*|}\right)$ stands for the geometric distribution with mean $|\mathcal{T}^*|$.

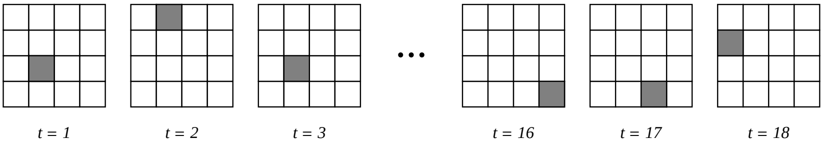


Figure 3.12: Representation of the cell being updated (gray) at time steps 1, 2, 3, 16, 17 and 18 according to the random independent update method.

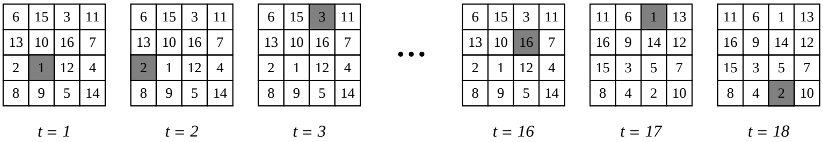


Figure 3.13: Representation of the cell being updated (gray) at time steps 1, 2, 3, 16, 17 and 18 as well as the update order according to the random order update method.

Random order update method All cells are updated consecutively in a random order. After every cell c_i in $|\mathcal{T}^*|$ is updated, the random order is shuffled. The random order used for the first 16 time steps is indicated for the tessellation at $t = 1$, $t = 2$, $t = 3$ and $t = 16$ in Figure 3.13. After the 16th update, a new order for update is chosen at random as can be seen from the two right most configurations depicted in Figure 3.13. Making use of the parameters introduced earlier, it follows that

$$\forall\ i\ : E_i < 2\ |\mathcal{T}^*|,\ F_i \sim \mathcal{H}\left(\frac{1}{|\mathcal{T}^*|}\right),$$

$$\forall\ t\ : G_t = 1\,.$$

Random cyclic update method At each time step a cell c_i is selected for update according to a predefined, fixed order, as opposed to the random order. Figure 3.14 is similar to Figure 3.13 except for the update order being the same for all the time steps of the simulation, such that it can be written that

$$\forall\ i\ : E_i = |\mathcal{T}^*|,\ F_i \sim \mathcal{H}\left(\frac{1}{|\mathcal{T}^*|}\right),$$

$$\forall\ t\ : G_t = 1\,.$$

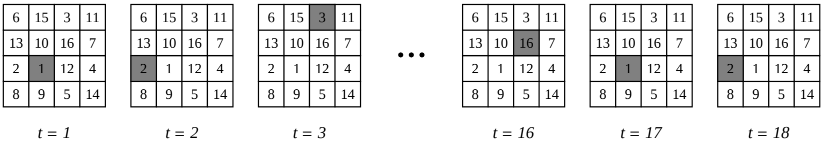


Figure 3.14: Representation of the cell being updated (gray) at time steps 1, 2, 3, 16, 17 and 18 as well as the update order according to the random cyclic update method.

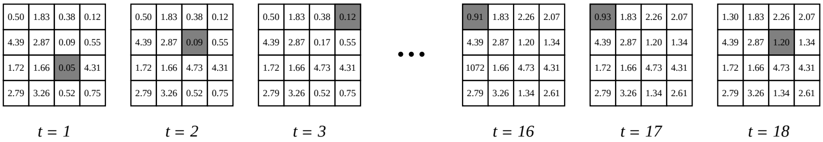


Figure 3.15: Representation of the cell being updated (gray) at time steps 1, 2, 3, 16, 17 and 18 as well as the exponentially distributed waiting time of every cell according to the exponential update method.

Exponential clocked update method Every cell c_i has a clock, which is specific for each cell and indicates when the cell is to be updated. The waiting time for update is exponentially distributed with a mean of one. To implement this, initially, every cell c_i is assigned a waiting time p_i according to the aforementioned exponential distribution. The cell that possesses the lowest value of p_i is updated first and afterwards a new p_i^* is selected at random from this exponential distribution which is then added to the old value, i.e. $p_i = p_i + p_i^*$. After this, the cell that now has the lowest p_i is updated, and this process continues until a preset number of time steps is reached. This is represented in Figure 3.15. Following the notations of the other update methods, it follows that

$$\forall\ i\ : E_i \geq 1\,,$$

$$\forall\ t\ : G_t = 1\,.$$

3.4. Pros and cons

It is important to state that the use of CAs does not exclude the use of other modeling paradigms, but is rather complementary to the use of those paradigms, especially for systems that consist of discrete components with local interactions. CA-based models can be seen as an alternative to PDE-based and other more classical models, providing researchers with a wider range of modeling tools and, in some complex cases, a solution to problems encountered with some of the more classical modeling methods. Furthermore, there exists a large number of researchers who are using a mixture of CA modeling and other techniques, including the development of hybrid models, to tackle challenging problems [64, 65, 66, 67, 68, 69, 70].

CAs have a number of characteristics that can be advantageous for mathematical modeling. First of all, the level of knowledge that the modeler has of the process is of importance. When the modeler has no knowledge of the underlying process at hand, the intractable dynamics can be translated into a CA transition function that reflects intuition and is easy to compute [38, 71]. Therefore, CAs have great potential to simulate natural phenomena when there is no other mathematical model available [72, 73]. Further, although the processes underlying the phenomenon might not be completely elucidated, the information a modeler has at hand can still be employed, since CAs lend themselves well for gray-box modeling (see Section 2.2). Finally, even when all necessary information is at hand, PDE-based modeling constitutes a roundabout approach if one considers that it first involves the derivation of a set of continuous equations from discrete mass balances after which these equations have to be discretized again as it is for most practical problems impossible to retrieve an analytic solution [22]. By making use of CAs to directly calculate the solution, the modeler can circumvent the latter tedious job.

A second important issue to take into account when developing a model is the choice of the level of description, be it microscopic, mesoscopic or macroscopic. Microscopic models mostly focus on processes taking place at the atomic level, i.e. between different molecules, but are not interesting for many chemical processes since the phenomena that are studied constitute rather scarce events at the atomic scale. Furthermore, with microscopic models, the state of the cells often represents an individual molecule. Obviously, this limits the system space that can be studied without resulting in excessively large computation times. On the other hand, a macroscopic description with classical kinetic equations, besides the fact that in most cases analytical solutions of the resulting equations are not at hand as was mentioned before, is often unable to capture the stochasticity of the underlying processes causing that some of the information, important to engineers, is not readily available [10, 74]. The mesoscopic approach is seen as a way to combine the macroscopic phenomenology with the stochastic character of the processes originating from the microscopic scale processes [75]. CAs are suited to construct these mesoscopic models, where the goal is not to describe a specific system, but rather to analyze how a combination of a small number of basic processes, very well accepted by researchers, might determine general features. The latter entails that cells are not to be associated directly with the individual atoms or molecules, but rather with a homogeneous grouping of the same type of atom or molecule. This means that atomic size effects are not accounted for, which is a simple approximation intended to capture the synoptic effects of the process at hand [76, 77].

Finally, the modular nature of CAs, which enables them to efficiently handle heterogeneity and/or complicated boundary conditions [78, 79], can be used to the modelers benefit. Moreover, it also makes them more immediately suitable for parallelization which may give very large increases in speed in comparison to more conventional models.

Nevertheless, there are some points of attention when using CAs. A major point is that the modeling framework for CAs is not as vast as for PDEs, for instance. Up to today, no general framework for designing and testing CA-based models for real-world problems and validating them using experimental data has been established [80]. This is partly due to the skepticism in the scientific community about the CA paradigm, but mainly because CA-based models are intrinsically difficult to validate. This is due to the absence of a direct link between model parameters and real-world physical variables and the intricate translation of a discrete time step to continuous time. Unfortunately, this lack of general guidelines according to which results are reported often results in incomplete reports of research efforts, making it difficult or even impossible to reproduce results. Finally, despite the fact that the possibility of constructing transition functions based on intuition is an advantage, it opens the door to misinterpretations. The latter meaning that the risk exists that researchers are reading too much into the simulated results and may draw incorrect conclusions.

4 Sensitivity analysis

4.1. Definition, uses and potential pitfalls

A possible definition of sensitivity analysis (SA) is that it is the study of how uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input (see Figure 4.1) [81]. A common uncertain type of input is a model parameter, especially when modeling complex natural systems [82, 83]. Therefore, one of the first steps taken by modelers when studying a newly developed model is to perform a SA, to obtain useful insights into the level of uncertainty contributed by each parameter.

SA methods have been mostly developed in the risk analysis realm. It allows to check how the risk is distributed among parameters and which assumptions to monitor in order to reduce risk. However, SA nowadays is an established practice in different (scientific) fields. Depending on the field and the specific method used, motivations for SA are [81]:

- Model corroboration. It can be checked whether the model is overly dependent on fragile assumptions.
- Research prioritization. Which input is most deserving of future analysis or measurement?
- Model simplification. Possibly, some inputs or parts of the model can be fixed or simplified. In this way, model complexity could be reduced by filtering out parameters that have a minor effect on the model outcome.
- Identifying critical or otherwise interesting regions in the space of the input factors. Identifying factors which interact and which may thus generate extreme values. This is important for system reliability.
- Prior to parameter estimation, to help set up the (actual or numerical) experiment in those conditions in which the sensitivity of the output to the factor to be estimated is the greatest.

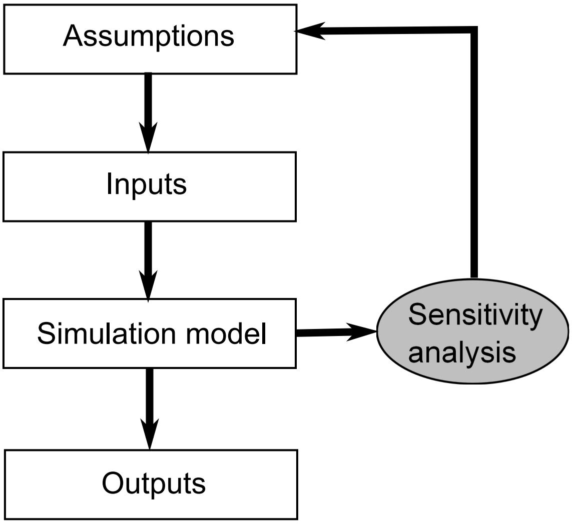


Figure 4.1: Schematic representation of the process of sensitivity analysis.

Further in this dissertation, SA will also be used to improve the parameterization of models (see Part III). The use of SA information in improving parameter estimation is not new, however, it is mostly used to guide (additional) experiments to gather data [84]. The main advantages here are that the sensitivity information is already available through the model exploration and that this information is suitable for different parameter sets, since the sensitivity information does not depend on the experimental data.

Despite the potential advantages an SA holds, there are some pitfalls if SA is at stake. The most important one is that the underlying purpose needs to be clearly defined. A researcher can perform a lot of different (statistical) tests to study a problem and not become any wiser afterwards. Secondly, a modeler needs to be aware that there exist different methods for different purposes. This means that it comes down to using the right technique in the right situation. More often than not the choice of method will be challenged before anything else. Finally, performing an SA when too many inputs or outputs are involved diminishes the value of the SA results. In what follows, an overview of different types of SA are presented.

4.2. Local and global sensitivity analysis

There exists a wide variety of SA techniques that all express how a small change in parameter (which will be the inputs for the models in this dissertation) value(s) influence(s) the output [81]. Most of the SAs met in literature are based on derivatives. Indeed, the derivative $\partial Y/\partial q_i$ of an output Y with respect to an input parameter q_i can be thought of as a mathematical definition of the sensitivity of Y for q_i. However, derivatives are only informative at the base point where they are computed and do not provide for an exploration of the rest of the parameter space Ω. Such an SA that does not attempt to fully explore the parameter space, but focuses on the sensitivity around a fixed point $\boldsymbol{q_0}$ is called a local sensitivity analysis.

The majority of SAs in chemistry and physics is performed locally, partly due to their usefulness for inverse problem solving [83]. Often, a local SA is performed by changing the value of one parameter at a given time while keeping all other parameter values constant and allows as such to detect the net effect of a single parameter on the model outcome [83, 85, 86]. The sensitivity H_i of the model output Y to a single parameter q_i is in this case computed as:

$$H_i = \left| \frac{\partial Y(q_1, ..., q_i, ..., q_b)}{\partial q_i} \right|_{\boldsymbol{q_0}}, \tag{4.1}$$

with b the number of model parameters. The simplest method to compute local sensitivities involves replacing the derivative in Eq. (4.1) by its finite difference approximation, so that Eq. (4.1) is rewritten as:

$$H_i = \left| \frac{Y(q_1, ..., q_i + \Delta q_i, ..., q_b) - Y(q_1, ..., q_i, ..., q_b)}{\Delta q_i} \right|_{\boldsymbol{q_0}}. \tag{4.2}$$

This approximation is only valid for a sufficiently small perturbation of q_i, because the nonlinearity of the model will become important if the perturbation factor $1 + \Delta$ is too large. However, Δ should also not be too small, as this would give rise to numerical inaccuracies [86].

The fact that a local SA only looks at the sensitivity at a defined base point, does not matter for linear systems, since the sensitivity at any point in the parameter space is the same. For nonlinear systems on the other hand, the results from a local SA do not tell anything about the sensitivity in the parameter space Ω except at $\boldsymbol{q_0}$. For that reason, more interesting when performing model exploration is a global SA as it can give an idea of the sensitivity over the full range of the parameters and where all parameters are varied at the same time, although this comes at the cost of a higher number of model runs needed. Some global SA techniques rely on averaging out a set of local SAs [87, 88], although many different types of global SA techniques exist, each of them with certain advantages depending on the case at hand. Table 4.1 gives a (non exhaustive) overview of some of the more popular types of global SA and their advantages and disadvantages.

Table 4.1: Overview of global SA techniques with advantages and disadvantages.

Technique	Advantages	Disadvantages
Regression-based	* Simple to code * No design required * Quick assessment	Works for (quasi) linear models
Rank-based regression and correlation	Works for nonlinear models	* Works for monotonic models * Conclusions do not translate to original model
Screening	* 10 < # inputs < 100 * Computationally intensive models * Analysis by groups of inputs	* Special design * Special sampling * Unclear interpretability of importance indices
Variance-based	* Clear interpretability of importance of indices * Interaction at any order * Analysis by groups	* Special design * Computationally expensive * Second order moment
Moment-independent	Captures changes in the entire distribution	* Specific design * Computationally expensive
Meta-modeling	Cheap functions to replace expensive models	* Maximal dimensionality (20–30) * Loss of parameter interpretability
Based on Monte Carlo sampling	* No design required * Model free * Fast to implement	Limited to first order effects

4.3. Choice of appropriate technique

The choice of method of SA is typically dictated by a number of problem constraints or settings. The most common ones are computational expense, correlated inputs, linearity of the model results, multiple outputs and whether or not the model itself is available or just a set of given data obtained with the model.

In deterministic models, the outcome for a specific set of parameters is essentially the same for identical initial conditions. Stochastic models, like the ones that will be used in this dissertation, on the other hand, have varying outputs between simulations, even if parameter values and initial conditions are identical [85]. Rather than comparing two single output values for sensitivity analysis, as is the case for deterministic models, two distributions of output values have to be compared. SA for stochastic models is often based on the mean of the different distributions of output values, although a better approach also accounts for the shape of the distributions, for instance by taking into account the variance [83].

Furthermore, since simulations with CA-based models are generally more time consuming than e.g. simulations with a PDE-based model, a screening method for SA is adopted here because such method has a relatively low computational cost when compared to other approaches. With screening methods, the objective is to identify which input variables are contributing significantly to the output uncertainty and roughly rank them, rather than exactly quantifying sensitivity. They are often employed for problems with long model evaluation times or reasonably high dimensionality.

Taking both arguments into account, the Elementary Effects (EE) method or method of Morris [88, 89, 90] is used throughout this dissertation as SA technique, because it is a variance-based screening method that is thus not overly computationally expensive.

4.4. Elementary Effects method

The EE method is an effective variance-based screening method that uses an average measure of the EE effect of each parameter by observing the effect on the output. Note that this is a quantitative method, but does not have a direct interpretation in terms of output variance. The design consists of $\mathcal{R}$ trajectories in the parameter space Ω. The range of each parameter is normally divided into a number of levels, creating a grid. A random point is chosen as a starting point. The design then moves a distance of Δq_1 in the direction of q_1, then Δq_2 in the direction of q_2 and so on for all b parameters. This creates a winding stairs trajectory of $b+1$ points. After the trajectory is constructed, a new random point is chosen and a new trajectory is constructed. This is repeated for $\mathcal{R}$ trajectories, at a total cost of $\mathcal{R}\,(b+1)$ model runs.

The sensitivity is calculated for each parameter as the mean of $\mathcal{R}$ elementary effects:

$$EE_i = \frac{Y(q_1, \ldots, q_i + \Delta q_i, \ldots, q_b) - Y(q_1, \ldots, q_i, \ldots, q_b)}{\Delta q_i}, \tag{4.3}$$

i.e.

$$\mu_i = \frac{\sum_{r=1}^{\mathcal{R}} EE_{i,r}}{\mathcal{R}}, \tag{4.4}$$

with r an index referring to the runs. The overall influence of a parameter q_i on the output is assessed using this μ_i. Often it is useful to compute μ_i^*, being the mean of $|EE_{i,r}|$, as it comes at no extra cost to calculate and resolves the issues encountered with non-monotonic models [91]:

$$\mu_i^* = \frac{\sum_{r=1}^{\mathcal{R}} |EE_{i,r}|}{\mathcal{R}}. \tag{4.5}$$

Another useful measure of nonlinearity and interaction is given by the standard deviation of the modulus of the elementary effects:

$$\sigma_i = \sqrt{\frac{\sum_{r=1}^{\mathcal{R}} (EE_{i,r} - \mu_i)^2}{\mathcal{R}}}. \tag{4.6}$$

The logic here is that if the response of the output to a given input were perfectly linear, the EE would be identical anywhere in the input space. In the nonlinear case the latter would not hold true. The importance of the different parameters can be assessed graphically using a (μ^*, σ)-plot, with the parameters closest to the origin, i.e. with low value for both μ^* and σ, being the least influential [91].

Nothwithstanding the fact that a few authors in literature report an SA for CA-based models [92, 93], the use of the EE method, although widely and successfully used in literature and recently also for stochastic biological models [94], for CA-based models is to the author's knowledge very limited.

PART II

CASE STUDIES

5 Introduction

Part II of this dissertation is all about exploiting the advantages of CAs and trying to overcome their limitations. Four different case studies receive ample treatment in this part. All case studies are distinct, both in the process that is modeled as well as in the aspects of CA modeling explored to develop the models. In this way, a range of possible applications and fields of research are presented next to a more detailed description on what modeling choice can be made to overcome a specific hurdle. The level of complexity of the developed model and modeling choices as well as the experimental data used increases from the first to the last case study.

Each of the chapters is organized similarly. First some details are given on the process that is modeled and the previous modeling efforts that have been made in this field. Next, the data acquisition is discussed after which the major part of each chapter deals with the development of the stochastic CA-based model and the modeling choices linked to this. Once the model is developed, experimental data are used to solve the inverse problem, i.e. to retrieve the values of the model parameters such that the simulated data come as close as possible to the observed data, also called parameterization of the model. Finally, some conclusions are drawn and future work is suggested with focus on the modeling choices made in that chapter.

With regard to the experimental data, time series are used to parameterize the CA-based models in all four case studies. A time series is a collection of observations of well-defined data items obtained through repeated measurements over time, usually in regular time intervals [95]. Time series occur naturally in many application areas such as economics, finance and medicine. Two main applications of time series data can be discerned. A first one is to obtain an understanding of the underlying forces and structure that produced the observed data. The second use is to fit a model and proceed to forecasting, monitoring or even feedback and feedforward control, which is also why they are used throughout Part II of this dissertation. The time series employed in Chapters 6 and 9 were gathered in the laboratory by the author, while the time series employed in Chapters 7 and 8 were made available to the author.

Chapter 6 deals with the reaction of calcium carbonate and hydrochloric acid. A CA-based model with only one parameter that incorporates diffusion and reaction is developed. In order to show that CAs can indeed be an alternative to PDEs, a PDE-based model with one parameter describing the same reaction is introduced. This chapter discusses the advantages and limitations of incorporating the effects of various processes into a single parameter. Further, the use of the block CA paradigm for this specific case is motivated and discussed. Finally, the parameterization of the developed model is done through an exhaustive search which was possible due to the presence of a single parameter. The experimental data employed in this chapter consist of a single time series of the decrease in mass due to the escape of formed CO_2 from the reaction mixture.

The second case study discusses the photocatalytic degradation of fluoroquinolones. The four key processes behind this phenomenon, i.e. photolysis, adsorption, desorption and photocatalysis, are incorporated in the CA-based model developed in Chapter 7. Four model parameters are employed, one for each key process. From this chapter, it is clear that there exists a relationship between the physical dimension of a tessellation cell, the physical time that corresponds to a discrete time step Δt and the diffusion coefficient. When, like is the case in Chapter 7, the diffusion coefficient is known, choosing a number of seconds that corresponds to a time step at the same time determines the physical size of a cell and vice versa. Like in Chapter 6, the system being modeled is well mixed such that the choice is made to model only a part of the entire system which is then afterwards scaled in order to minimize computation times. However, here sparse arrays are preferred, since the phenomenon being modeled is more complex and the use of a block CA would result in a continuous scanning of the entire tessellation in order to retrieve the small number of cells that are of interest. The latter would result in a severe computational effort. Periodic boundaries are used to simulate the presence of a larger system. In order to still respect the law of mass conservation, having abandoned the block CA paradigm, an asynchronous update method is adhered to. Finally, the results of an SA are used to guide the parameterization of the CA-based model which is done via grid search using experimental data from a lab-scale batch reactor. Here, five different time series of experimental data are at hand that all describe the same process, but with different initial concentrations of fluoroquinoles to degrade. All five time series are used to retrieve the optimized parameter set.

In Chapter 8, a CA-based model is proposed to describe oil migration in chocolate-coated confectionery. Like in Chapters 6 and 7, a 2D tessellation is used. However, this time the tessellation represents a cross-section of the complete system under investigation, rather than a small part. The latter has an influence on the choice of tessellation resolution as well as on the choice of boundary conditions, both of which are elaborated on in this chapter. One of the four model parameters to be optimized is Δt, the physical time corresponding to one discrete time step. This is due to the fact that there is no value available for the diffusion coefficient D under the circumstances described in Chapter 8. After parameterization of the CA-based model, an effective diffusion coefficient D for this specific case is calculated. Like in Chapter 7, more than one time series of data (two in this case) is used to retrieve a single optimal parameter set. However, in this case the time series do not constitute the same process at different concentrations, but two time series originating from two different processes. Therefore, the optimized parameter set has to correctly give rise to two different types of output simultaneously. Finally, it is mentioned that the choice to apply the transition function in an asynchronous manner is based on the fact that synchronous updating would result in an excessive number of conflicting movements during the diffusion part of the transition function as such influencing the parameterization to an unacceptable extent for further interpretation.

Finally, Chapter 9 introduces a 3D CA-based model with the incorporation of the key processes of pitting corrosion. In contrast with the first three case studies, a 3D tessellation is employed since the experimental data call for it: three dimensions are needed when the surface and the depth of corrosion pits have to be modeled simultaneously. The domain, like is also done in Chapters 6 and 7, represents only a part of the entire system. However, here a combination of periodic and fixed boundaries is used since the system under study is not a well-mixed reaction solution, but contains an interface of two different components. Further, again Δt is introduced as a parameter since no diffusion coefficient is known for the experimental conditions at hand. Although the situation in this case study is similar to the one presented in Chapter 8, the update of the state of the cells of the tessellation is not performed in an asynchronous manner, the reason for this being twofold. Firstly, the number of update conflicts in this specific case is limited and so the influence on the optimal parameter values is small. The second reason for choosing a synchronous updating mechanism is the reduction in computation times it offers, which is of utmost importance when working with 3D models. Finally, the results of an SA are used to simplify the parameter estimation. Via a grid search in two steps, the inverse problem is solved using three different time series that each describe a different type of output. It is mentioned here that the time series of experimental data used in this chapter had to be derived from a set of micrographs and another, not directly employable, time series of data.

6 Reaction kinetics

In this chapter, a stochastic CA-based model for modeling physico-chemical reactions is designed, applying it as illustration to a well-known reaction, namely the reaction of calcium carbonate ($CaCO_3$) with an aqueous solution of hydrochloric acid (HCl). The main goal of this chapter is to illustrate that a CA-based model can be an alternative for a PDE-based model. Special attention is given to the workability of the CA-based model, incorporating, a.o., the block CA paradigm, to create a practical modeling tool for performing calculations with real data instead of developing a CA-based model that merely gives qualitative results, as was mostly done in the past [15, 33]. The model takes into account the changes in concentration due to diffusion as well as reaction. Experimental data were gathered in the laboratory and the CA-based model was parameterized using these data. The optimization is performed through an exhaustive search of the solution space since the interest lies in an approximate solution to the problem. Furthermore, a PDE-based model, for which the system of PDEs is deduced from a CA, is parameterized using the same experimental data in order to compare both models.

6.1. The phenomenon

Chemical kinetics or reaction kinetics is the study of rates of chemical processes, meaning that it investigates how different experimental conditions can influence the speed of a chemical reaction. Further, it yields information about the reaction's mechanism and transition states, as well as the construction of mathematical models that can describe the characteristics of a chemical reaction. In 1864, Waage and Guldberg [96] pioneered the development of chemical kinetics by formulating the law of mass action, which states that the speed of a chemical reaction is proportional to the quantity of the reacting substances. Chemical kinetics deals with the experimental determination of reaction rates from which rate laws and rate constants are derived. The reaction studied in this chapter is that of $CaCO_3$ with HCl, which is used to produce the salt $CaCl_2$ that finds applications in brine for refrigeration plants, ice and dust control on roads, and desiccation.

6.2. Data acquisition

The reaction of $CaCO_3$ with HCl can be written as

$$CaCO_{3(s)} + 2\,HCl_{(aq)} \rightarrow CaCl_{2(aq)} + H_2O_{(l)} + CO_{2(g)}. \tag{6.1}$$

The reaction given by Eq. (6.1) was conducted in the laboratory at ambient temperature and pressure to obtain a time series of the cumulative mass CO_2 that escaped from the solution. For that purpose, 7.5 g of $CaCO_3$ (GPR RECTAPUR, 99 % purity) was added to $35.6\,10^{-3}$ l distilled H_2O in a measuring beaker open to the air. The $CaCO_3$ dissolved until the liquid was saturated, while the remaining undissolved $CaCO_3$ sank to the bottom of the beaker. To this mixture, $14.4\,10^{-3}$ l of a 23 % (w/w) HCl solution (AnalaR NORMAPUR) was added. The density of this solution was 1100 g/l, which means that there was 3.64 g (0.0998 mol) HCl present at the beginning of the reaction, next to 0.075 mol of $CaCO_3$. Taking into account that two moles of HCl are required to gasify one mole of $CaCO_3$, $CaCO_3$ was in excess.

As soon as the $CaCO_3$ and HCl solution came into contact with one another, gaseous CO_2 was formed that immediately escaped from the open beaker. The weight loss due to this gas formation was measured by means of an analytical balance (Mettler-Toledo, ± 0.0001 g). The solution was continuously stirred in order to keep the reaction going, promote the dissolving of additional $CaCO_3$ and allow the formed CO_2 to easily escape from the stirred solution. The weight loss of CO_2 was measured for 565 seconds since by then the weight loss was no longer significant. In the beginning a reading was done every 5 or 10 seconds and towards the end of the experiment every 100 seconds. The theoretical stoichiometric maximum weight loss due to CO_2 formation is 3.3 g.

During the execution of the experiment, it was noticed that other sources of weight loss occurred like evaporation to the open air, as well as adhesion of fluid on the plastic beaker where it could evaporate more rapidly. These additional sources of weight loss were quantified by means of a blank experiment under the same conditions, and added up to 0.80 g at the end of the experiment. Corrections to the model output were made, making use of the quantified additional weight loss, by adding this additional weight

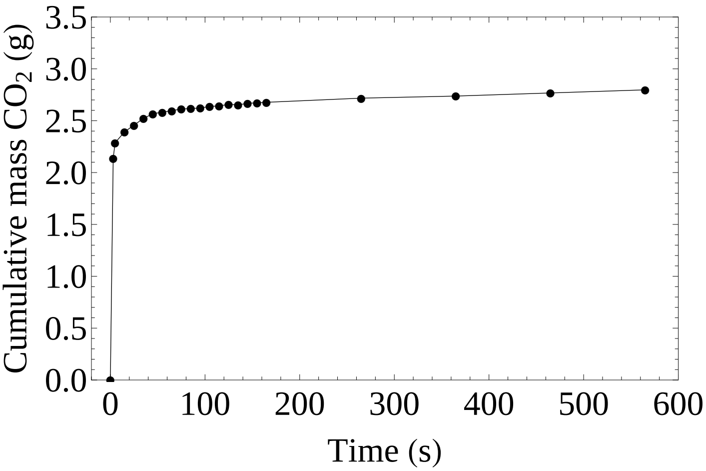


Figure 6.1: Cumulative loss of CO_2 in function of time

loss to the simulated weight loss by the model to give a total simulated weight loss. The experimental loss of CO_2 in function of time is given in Figure 6.1.

6.3. Gasification of calcium carbonate

6.3.1. The CA-based model

In this chapter a stochastic CA-based model is developed according to Definition 3. A tessellation of $\mathbb{R}^2$ that consists of squares is used, because it has the most straightforward implementation and suffices to illustrate the validity of the model that will be constructed. The size of the tessellation $\mathcal{T}^*$ used is 99×99 cells. Further, the Moore neighborhood function is used and a set of states $S = \{H_2O, HCl, CaCO_3\}$ is adhered to. For reasons of simplicity and workability of the model, no additional state for the CO_2 or $CaCl_2$ molecules is introduced, the former because it is assumed that they escape immediately from the medium and are therefore never present in the tessellation and the latter because they are completely dissolvable salts that are non-reactive.

To initialize the CA at the beginning of a simulation, every $c_{i,j} \in \mathcal{T}^*$ is assigned a state according to the molar ratios of the reactants and water in the recipient at the start of the experiment described in Section 6.2.

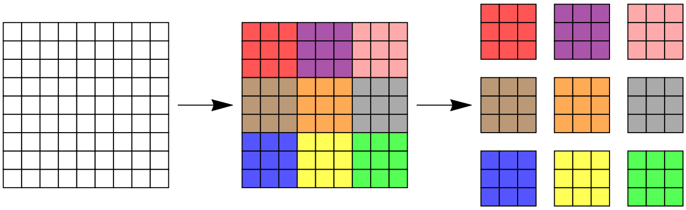


Figure 6.2: Splitting a tessellation $\mathcal{T}^*$ in blocks of three by three cells.

After the initialization of the tessellation a scaling factor κ, which links the number of cells of the tessellation in state $CaCO_3$ at any given time to a physical amount of $CaCO_3$, is calculated. Therefore, the total number of cells of the tessellation in state $CaCO_3$ at $t = 0$ is compared to the number of $CaCO_3$ molecules present at the beginning of the experiment. The latter is calculated by determining the number of molecules present in 0.075 mol $CaCO_3$ (7.5 g $CaCO_3$), which is the exact amount of $CaCO_3$ present at the beginning of the experiment, making use of Avogadro's number ($6.022\,10^{23}$), resulting in $4.517\,10^{22}$ $CaCO_3$ molecules. For the tessellation size considered here, the scaling factor κ is equal to $1.711\,10^{20}$. Thus, the tessellation represents only a part of the entire system under study and is therefore more a calculation tool rather than an accurate representation of a system.

The CA-based model is implemented by relying on the block CA paradigm (see Section 3.3). This choice is made here because it allows the transition function to be applied in a synchronous manner while still obeying the law of mass conservation. Here, the tessellation is split into non-overlapping blocks of three by three cells $c_{i,j}$, $c_{i+1,j}$, $c_{i-1,j}$, $c_{i,j+1}$, $c_{i,j-1}$, $c_{i+1,j+1}$, $c_{i+1,j-1}$, $c_{i-1,j+1}$, $c_{i-1,j-1}$, where $c_{i,j}$ is the central cell of the block (see Figure 6.2).

The transition function Ψ of the CA-based model falls apart into two parts which describe diffusion [1] and reaction, respectively. Diffusion is simulated by redistributing the state of the cells within every block of nine cells (see Figure 6.3), which is plausible since the system under study is well-mixed. This redistribution may happen with a probability $P_D \in]0,1[$ or can occur at every time step, i.e. $P_D = 1$. A second step, simulating reaction, consists of an evaluation of the central cell $c_{i,j}$ of every block of nine cells. The

---

[1] Throughout this chapter diffusion is employed to refer to the movement of molecules in the system, although in reality the driving force for movement in the system is advection.

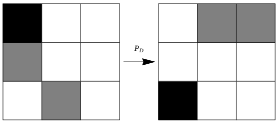
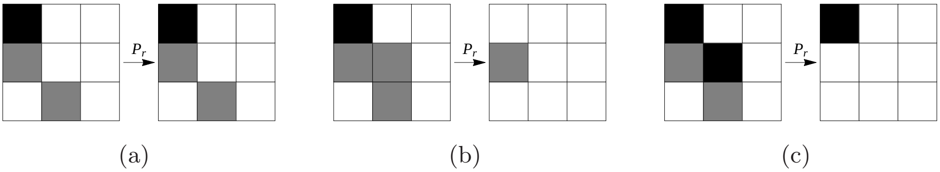


Figure 6.3: Simulation of diffusion by redistributing the state (H_2O (white), HCl (gray) and $CaCO_3$ (black)) of the cells within every block of nine cells.



(a) (b) (c)

Figure 6.4: Reactions that can occur if the central cell is (a) H_2O (white), (b) HCl (gray) or (c) $CaCO_3$ (black).

molecule in $c_{i,j}$ will react with a probability P_r, which is a key model parameter representing the reaction rate constant at which Eq. (6.1) takes place. If reaction is allowed to occur, then there are, based on the stoichiometry of Eq. (6.1), several pathways, which are also illustrated in Figures 6.4(a) – 6.4(c):

1. $c_{i,j}$ *contains a water molecule*: as a water molecule is non-reactive, nothing happens.

2. $c_{i,j}$ *contains a HCl molecule*: if another HCl molecule as well as a $CaCO_3$ molecule are present in the Moore neighborhood, reaction takes place with probability P_r and the three cells involved in the reaction turn to water, while CO_2 is formed and immediately escapes. If multiple possibilities for reaction exist, one combination is chosen at random.

3. $c_{i,j}$ *contains a* $CaCO_3$ *molecule*: if two HCl molecules are present in the Moore neighborhood, reaction takes place with probability P_r and the three cells involved in the reaction turn to water, while CO_2 is formed and immediately escapes. If more than two HCl molecules are present, two are selected at random.

Despite the synchronous updating of the different three by three blocks, only one cell, i.e. the central cell, is evaluated for the reaction part of Ψ, such that the applied update method constitutes an asynchronous update

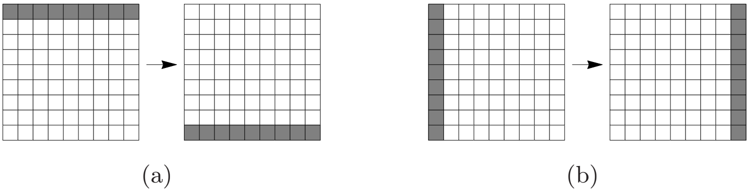

(a) (b)

Figure 6.5: Shifts that take place with the block CA-based model: (a) shift of upper row to the bottom and (b) shift of the utter right column to utter left.

method for this part of the transition function. To ensure that at every time step another cell is considered as central cell, a shifting mechanism is incorporated (see Section 3.3). The specific rule applied here is that after every time step the first row of the tessellation is removed and placed at the bottom (see Figure 6.5(a)) and the utter right column is transferred to the utter left (see Figure 6.5(b)).

Finally, the choice is made to set $P_D = 1$ for all simulations, making the diffusion part of the transition function deterministic. This implies that the stochasticity associated with the diffusion process is incorporated into P_r. Figure 6.6 shows a flow chart of the CA-based model developed in this chapter.

6.3.2. The PDE-based model

CAs can be an alternative, as mentioned earlier, for PDEs in the study of spatio-temporal processes. Therefore a continuous PDE-based model is implemented and parameterized to investigate whether a higher correlation can be found with regard to the observed data than with the CA-based model. The system of PDEs is deduced according to the principles described by Reichenbach *et al.* who show that a spatially discrete, stochastic system can be transformed into a system of stochastic PDEs, which can, in some cases, aptly capture a system's behavior in the continuum limit [97].

In agreement with the set-up of the CA-based model, a finite, square tessellation $\mathcal{T}^*$ is presumed in which every cell $c_{i,j}$ is either occupied with a $CaCO_3$ or a HCl molecule or is empty. Reaction is only allowed between the nearest neighbors, which are determined by the neighborhood function $N(c_{i,j})$ under consideration. Further, all molecules are given a mobility γ,

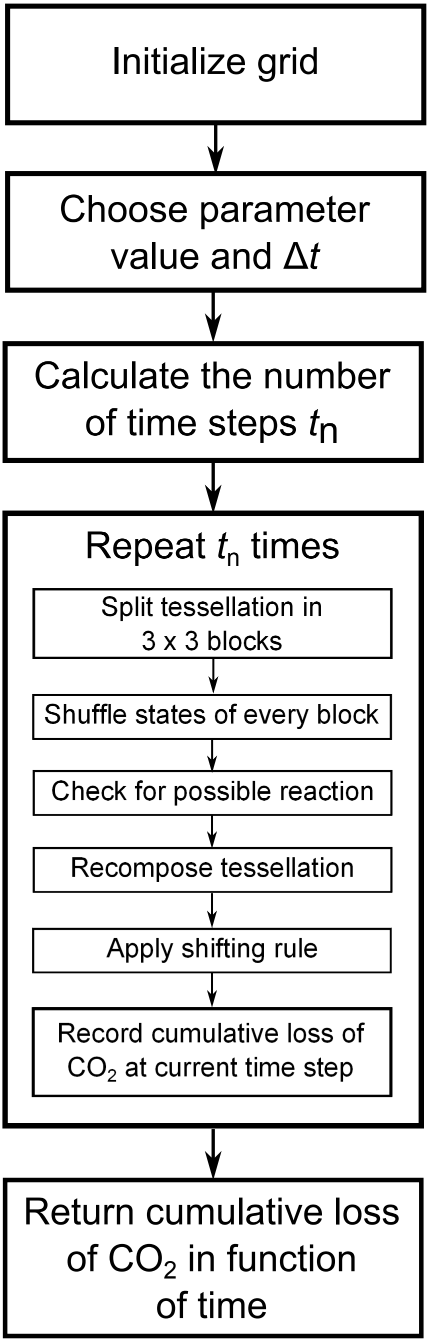


Figure 6.6: Flow chart of CA-based model for reaction kinetics.

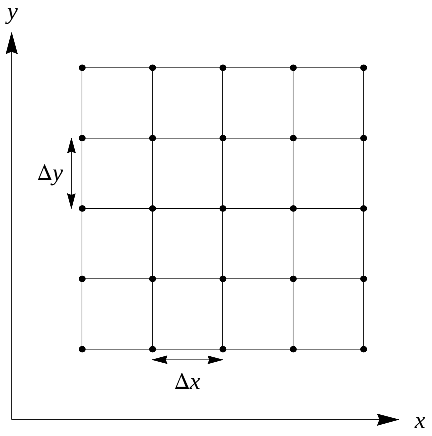


Figure 6.7: Representation of a spatially discrete system.

the speed with which molecules can move between neighboring cells. Note that this mobility γ has the same meaning as the parameter P_D in the CA-based model.

Let d denote the dimensionality, then the macroscopic diffusion coefficient D is defined as [97]:

$$D = \gamma \ d^{-1} \ |\mathcal{T}^{*}|^{-2/d} \, . \tag{6.2}$$

Here, D is assigned the value 10^{-9} m^2 s^{-1}, a value found in literature for similar conditions [97, 98]. If $\tau = 2d$ is defined as the number of closest neighbors of every cell $c_{i,j}$, P_r^* as the probability of reaction, the concentration of HCl and $CaCO_3$ at time t and position $\boldsymbol{r} = (x, y)$ as $X(\boldsymbol{r}, t)$, respectively, $Y(\boldsymbol{r}, t)$, and the distance between two neighboring cells in the x direction as Δx and in the y direction as Δy (see Figure 6.7), then it is possible to obtain a time evolution of the average value of $X(\boldsymbol{r}, t)$ and $Y(\boldsymbol{r}, t)$ for different neighborhood functions. In order to take a similar neighborhood as for the CA-based model, the Moore neighborhood will be considered and thus $X(\boldsymbol{r}, t)$ is defined as follows:

$$\frac{\partial X}{\partial t} = \frac{1}{\tau}\Big[\text{Diffusion} + \text{Reaction}\Big] . \tag{6.3}$$

The diffusion (reaction) term is the summation of the diffusion (reaction) of the constituents of the cell under consideration with each of its nearest neighbors. Therefore, a coefficient $\frac{1}{\tau}$ is included to average over the neighbors of a given cell $c_{i,j}$. The diffusion of HCl between x and $x-\Delta x$ is given by

$$2\,\gamma\,(X(x-\Delta x,y,t)-X(x,y,t)), \tag{6.4}$$

with γ being dependent on D as indicated by Eq. (6.2) and the addition of the coefficient 2 to account for the fact that diffusion can occur in both directions. When repeating this reasoning for all neighbors, the diffusion term of Eq. (6.3) can be written as

$$\begin{aligned}\text{Diffusion} = 2\,\gamma\Big(&X(x-\Delta x,y,t)+X(x+\Delta x,y,t)+X(x,y-\Delta y,t)\\ &+X(x,y+\Delta y,t)+X(x-\Delta x,y+\Delta y,t)+X(x+\Delta x,y+\Delta y,t)\\ &+X(x-\Delta x,y-\Delta y,t)+X(x+\Delta x,y-\Delta y,t)-8\,X(x,y,t)\Big).\end{aligned} \tag{6.5}$$

The second term of Eq. (6.3) is the reaction term. Two molecules of HCl and one molecule of $CaCO_3$ are needed for reaction, according to Eq. (6.1). When we consider reaction between HCl at (x,y) and at $(x-\Delta x,y)$, $CaCO_3$ can be taken from one of the seven remaining sites in the Moore neighborhood, resulting in seven possible ways for reaction to occur

$$\begin{aligned}
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x+\Delta x,y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x+\Delta x,y-\Delta y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x-\Delta x,y-\Delta y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x,y-\Delta y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x+\Delta x,y+\Delta y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x-\Delta x,y+\Delta y,t)\\
&-\;P_r^*\,X(x,y,t)\,X(x-\Delta x,y,t)\,Y(x,y+\Delta y,t)\,,
\end{aligned}$$

where the minus sign accompanying P_r^* indicates that HCl is consumed during reaction. This can be repeated for the reaction possibilities of HCl at (x,y) with HCl at one of the seven remaining neighbors in the Moore neighborhood besides $(x - \Delta x, y)$, each of them again generating seven reaction terms for the reaction with $CaCO_3$ at one of the other seven sites.

When we consider $\Delta x \to 0$, as well as $\Delta y \to 0$ and therefore $\Delta r = |\mathcal{T}^*|^{-1/d} \to 0$, $\mathbf{r}$ can be treated as a continuous variable which means that the Taylor expansion is justified:

$$X(\mathbf{r} \pm \Delta r, t) = X(\mathbf{r}, t) \pm \Delta r\, X'(\mathbf{r}, t) + \frac{1}{2}\Delta r^2\, X''(\mathbf{r}, t)\,, \tag{6.6}$$

upon dropping higher order terms. When Eq. (6.6) together with $\tau = 4$ (for a two-dimensional system) is considered, Eq. (6.3), after substituting the diffusion and reaction terms given by Eq. (6.5) and the different reaction parts, can be rewritten in the following manner:

$$\frac{\partial X}{\partial t} = \gamma\, \Delta r^2 \left(\frac{\partial^2 X}{\partial x^2} + \frac{\partial^2 X}{\partial y^2}\right) - 14\, P_r^*\, X(x, y, t)^2\, Y(x, y, t)\,. \tag{6.7}$$

By applying the same procedure, a similar equation can be obtained for $\partial Y / \partial t$ and taking into account the fact that $\gamma\, \Delta r^2 = 2\, D$ and by substituting X and Y by respectively [HCl] and $[CaCO_3]$, the following system of PDEs is obtained for a Moore neighborhood:

$$\begin{cases} \dfrac{\partial[\mathrm{HCl}]}{\partial t} = 2\, D\Big(\dfrac{\partial^2[\mathrm{HCl}]}{\partial x^2} + \dfrac{\partial^2[\mathrm{HCl}]}{\partial y^2}\Big) - 14\, P_r^*\, [\mathrm{HCl}]^2[\mathrm{CaCO_3}], \\[2ex] \dfrac{\partial[\mathrm{CaCO_3}]}{\partial t} = 2\, D\Big(\dfrac{\partial^2[\mathrm{CaCO_3}]}{\partial x^2} + \dfrac{\partial^2[\mathrm{CaCO_3}]}{\partial y^2}\Big) - 7\, P_r^*\, [\mathrm{HCl}]^2[\mathrm{CaCO_3}]\,. \end{cases} \tag{6.8}$$

The form of System (6.8) is in accordance with what can be found in literature for reaction-diffusion processes, whereby the reaction terms are specific for the reaction at stake [33, 48].

6.4. Parameterization

The probabilities P_r and P_r^* of reaction for the CA- and PDE-based models respectively, are parameters belonging to the interval $[0, 1]$. Because there is only a single parameter in both cases to be optimized and the interest lies only in an approximate value for P_r and P_r^*, it is opted to perform an exhaustive search for values of P_r and P_r^* with a limited number of decimal places. Further, it also stands without saying that the obtained optimized values for P_r and P_r^* are only valid for the conditions at which the experiment was conducted. If the model results are desired at for instance a different temperature, experimental data at this temperature has to be collected.

6.4.1. The CA-based model

At every time step of the evaluation of the CA-based model, the simulated loss of HCl molecules is registered. This loss can be linked to the number of CO_2 molecules formed and, together with the scaling factor κ mentioned in Section 6.3 as well as the molecular weight of CO_2, gives the simulated loss of CO_2 as a function of the time step. The experimental data is given as a cumulative loss of CO_2 in function of time. This means that a decision has to be made with regard to the relationship between the physical time and a discrete time step. This is done by choosing the number of seconds that equals one time step of the stochastic CA-based model. If one time step is set as 0.01 s, the cumulative loss of CO_2 after 100 time steps is linked to the observed weight loss after 1 s. The influence of this choice is embedded in the parameter P_r. It holds that a time step equaling a larger number of seconds, corresponds to a higher value of P_r in order to reach the same result.

The simulated CO_2 loss is calculated, using values of P_r ranging from 0 to 1 in steps of 0.001. The choice for an accuracy of 0.001 is based on the fact that the use of a smaller step size does not result in a significant decrease in root mean squared error (RMSE) between the experimental data and the simulated ones. The optimal value for P_r is chosen as the one that gives rise to the lowest RMSE. Because of the stochasticity of the CA-based model, the calculations are performed in tenfold, resulting in an average optimal value for P_r of 0.907, with a corresponding RMSE of 0.388. Figure 6.8 shows both the experimental data and the simulated data with the optimized value of P_r.

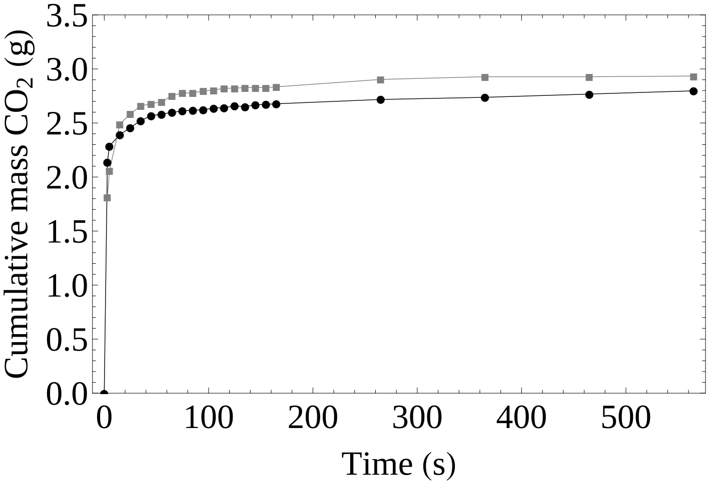


Figure 6.8: Simulated (▪), with CA-based model, and observed (•) cumulative mass loss of CO_2 in function of time.

6.4.2. The PDE-based model

The system of PDEs, described by (6.8), is solved numerically for values of P_r^* ranging from 0 to 1 in steps of 0.0001. In the case of the PDE-based model, a higher accuracy in comparison to the simulations with the CA-based model offers a significant improvement in the obtained RMSE. Here, the equations of the system of PDEs are expressed in physical time units and hence, there is no need for assuming a relationship between physical time and a discrete time step. The RMSE is again used as criterion to find an optimized value of P_r^* for which the simulations with the PDE-based model correspond best to the observed data. Since no stochasticity is involved, a repetition of the calculations is unnecessary and the P_r^* for which the lowest RMSE is encountered, is 0.0103. The corresponding RMSE amounts to 0.869. The simulated mass of CO_2 using the PDE-based model with the aforementioned value of P_r^*, as well as the observed measurements used in the parameterization procedure are represented in Figure 6.9.

Comparing the fit with the CA- and the PDE-based model by means of Figures 6.8 and 6.9, it can be seen that both models produce results that are simular to the experimental data and to each other. It is difficult to distinguish between the simulated results of the two models, except for the first three data points. It can be seen, that here simulated loss with the CA-based model fits the experimental data better than for the PDE-

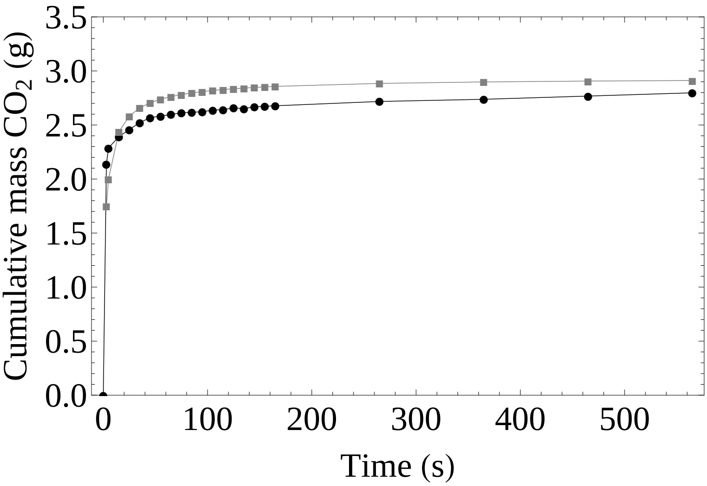


Figure 6.9: Simulated (■), with PDE-based model, and observed (●) cumulative mass loss of CO_2 in function of time.

based model. Further, it can be noticed that the optimized value for P_r is about 90 times bigger than that of P_r^*. This is due to a number of reasons. First of all, as mentioned before, the CA-based model is asynchronous, making that in every single time step P_r only has an impact on one ninth of all cells, while P_r^* influences all molecules simultaneously. Secondly, both parameters cannot easily be compared, since they are not completely the same: P_r^* is a pure reaction probability, whereas P_r also has the effect of diffusion as well as that of the choice of what physical time corresponds to one discrete time step embedded in it.

6.5. Conclusions

In this chapter, a CA-based model was presented to describe the reaction of $CaCO_3$ and HCl. This reaction was performed in the laboratory and the cumulative loss of CO_2 in function of time was registered. The CA-based model incorporates diffusion and reaction and gives a simulated loss of CO_2 in function of a discrete time step. The key parameter in this model was optimized using an exhaustive search, such that the RMSE between the simulated weight loss and the observed weight loss was minimal and that the fitted relationship had a high correlation to the observed data. Further, also a system of PDEs was implemented, starting from a set of CA rules, to compare the CA-based model to the more established continuous PDE-based models. After fitting the PDE-based model to the observed data, it could be seen that the simulated results with the models resemble both each other and the experimental data, with the CA-based model performing best. Nevertheless, Figures 6.8 and 6.9 show that there is still room for improvement. The use of just one model parameter in this case is probably an oversimplification of the process at hand. A more elaborate and complex model will potentially improve the fit of the simulated data to the experimental data. Further, it might also be necessary to consider more physical processes in the construction of the transition function of the CA-based model. One of these processes is the partial dissolution of CO_2 in the reaction solution. The latter implies that instead of assuming the instantaneous escape of CO_2, a delay factor is introduced, which dictates how long CO_2 remains in the reaction solution before escaping it.

This chapter not only shows that CA-based models are indeed an alternative for their PDE-based counterparts, but moreover, that a PDE-based model can be deduced starting from a set of CA rules. Further, by choosing to model a relatively simple reaction and by merging the effect of several parameters, the proposed CA-based model only has one parameter. On the one hand, this immensely facilitates the parameterization of the model to the point where an exhaustive search is possible. On the other hand it also blurs the interpretability of the optimized value of the parameter, making it difficult to attribute an unambiguous physical meaning to the result.

Another point to address here is that the tessellation used in this chapter does not represent the complete physical domain of the beaker in which the reaction was performed. The tessellation in this case is rather a small part of a much larger system that is used for calculations after which the results are scaled using the scaling factor κ. Although in theory it would be possible to construct a tessellation that accurately represents the complete physical system, in practice this would result in unnecessary calculations and excessive computation times. Especially since the CA-based model deals with a well-mixed solution where the reactions that are occurring are the same in every part of the system. The fact that we are dealing with a well-mixed system also justifies the use of the block CA, which speeds up calculations, facilitates implementation, obeys mass conservation laws and eliminates the need for boundary conditions.

7 Photocatalysis

Fluoroquinolones (FQs) are a family of synthetic broad-spectrum, antibacterial compounds, with moxifloxacin (MOX) being one of the most important members. Although very efficient and active, they are incompletely metabolized during human therapy and traditional wastewater treatment plants cannot remove them completely, so that they accumulate in the environment. Research is therefore directed towards novel methods, such as titanium dioxide (TiO_2) assisted photocatalysis, to achieve complete mineralization of organic contaminants. Despite the great potential of photocatalysis and the ongoing research in this field, the reaction mechanism behind photocatalysis is still not completely elucidated and the oxide surfaces of the TiO_2 photocatalyst are more complex than anticipated, which hampers the development of a mathematical model for describing photocatalytic degradation. In this chapter, a stochastic CA-based model is proposed to simulate the mechanism behind photocatalysis. The model is validated and evaluated through SA. Subsequently, it is parameterized using experimental data from a lab-scale batch reactor.

7.1. The phenomenon

FQs are a family of synthetic broad-spectrum, antibacterial compounds that are widely used for both human medicine and livestock operations [99, 100]. The most popular quinolone antibiotic in Europe is ciprofloxacin (CIP) (see Figure 7.1(a)), which is a potent second-generation fluoroquinolone that kills bacteria by interfering with the enzymes that cause DNA to rewind after being copied and which is used to treat, a.o., skin and urinary tract infections. However, the increased use and exposure to CIP can increase bacterial resistance so that the search for new compounds is ongoing. One of those new FQs is MOX (see Figure 7.1(b)), an upcoming fourth-generation FQ, which is at the moment the last resort when all other antibiotics have failed [100].

Although very efficient and active, FQs are incompletely metabolized during human therapy and are consequently partially (> 50 %) excreted in their

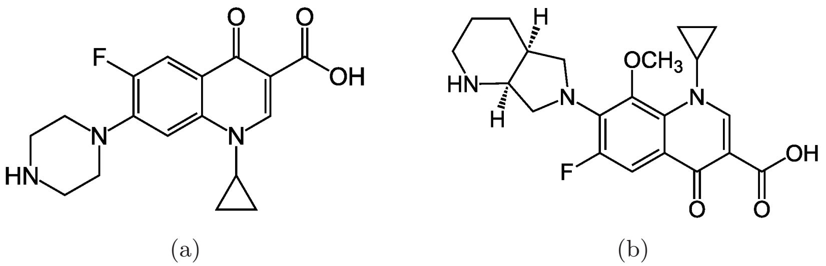


Figure 7.1: Molecular structure of (a) CIP and (b) MOX.

pharmaceutically active form, ending up in the municipal sewer [99, 101]. Due to their limited biodegradability, widespread use and incomplete removal in wastewater treatment plants, they accumulate in the environment and cause adverse effects on (aquatic) organisms [99, 101]. Therefore, FQs and other recalcitrant organic pollutants continue to increase in air and wastewater flows, which does not only pose an environmental problem, but as environmental laws and regulations are becoming more stringent, also creates an economic necessity to degrade these synthetical compounds which are resistant to conventional treatment methods [101, 102, 103, 104].

As such, research is directed towards advanced oxidation processes (AOPs) that rely on the generation of highly reactive radical species such as $HO^{\cdot}$ by using solar, chemical or other forms of energy, to obtain complete mineralization of organic pollutants. Among the AOPs, TiO_2-assisted photocatalysis is one of the most promising methods [101, 103, 105, 106, 107, 108]. Titanium has proven to be a good photocatalyst for it gives an excellent combination of photoactivity and photostability, i.e. it does not change upon exposure to light, while being well available and low in cost [102, 103, 104, 109]. When irradiated with UV light in the presence of O_2, TiO_2 catalysts generate highly reactive radical species that can convert organic compounds into CO_2, H_2O and mineral acids (see Eq. (7.1)), rendering them harmless:

$$\text{organic contaminant} \xrightarrow[\mathrm{TiO_2}]{\mathrm{O_2},\, hv} \text{intermediates} \rightarrow \mathrm{CO_2} + \mathrm{H_2O} + \text{mineral acid(s)}. \tag{7.1}$$

Despite the great potential of photocatalysis and the numerous studies in this field, the reaction mechanism behind photocatalysis is still not completely elucidated and the surface structure of the TiO_2 photocatalyst is more complex than anticipated, which hampers the development of a mathematical model describing photocatalysis [101, 103, 109]. Therefore, a stochastic CA-based model is proposed in this chapter to provide an alternative to the time-consuming experimental analyses and to gain further insight into the mechanism behind photocatalysis.

7.2. Data acquisition

For the parameterization of the CA-based model, experimental data were obtained according to the method described by Van Doorslaer *et al.* [105]. Experiments were performed using a lab-scale batch reactor (see Figure 7.2) which was kept at 298 ± 1 K and was equipped with a UV-A light source (485 $\mu W/cm^2$, 300-440 nm with main peak at 365 nm). The reactor was filled with a buffered solution to maintain a pH of 7, the organic pollutant to be degraded and the photocatalyst TiO_2 (if only photolysis was desired, the photocatalyst was omitted). The reaction mixture was, prior to irradiation, stirred under darkness and sparged with pure oxygen for 30 min so that an adsorption equilibrium was reached, after which the UV-A lamp was switched on and photocatalytic degradation took place (if TiO_2 was present).

The photocatalytic reactor, screened from natural light, is assumed to be well mixed and can therefore be considered as a batch reactor. At predefined times during the photocatalytic degradation, samples were taken, filtered and analyzed by a liquid chromatograph coupled to a photodiode array detector, as such measuring the concentration of organic pollutants. More details about this procedure can be found in Van Doorslaer *et al.* [105].

The experimental data used in this chapter were obtained from solutions with 5 g/l TiO_2 and different initial concentrations of MOX, being 5, 10, 15, 20 and 25 mg/l for photocatalysis and an initial MOX concentration of 15 mg/l and no TiO_2 for photolysis, i.e. the direct degradation through UV light [105]. Depending on the experiment, four up to eight measurements at different points in time were taken. From these measurements,

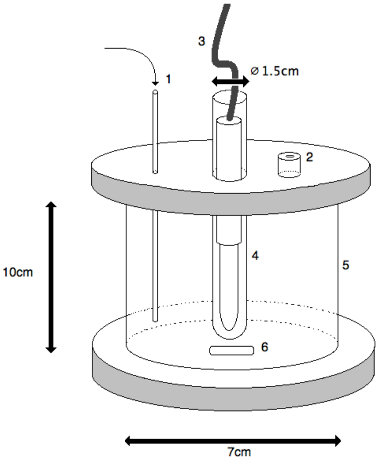


Figure 7.2: Lab-scale batch reactor with (1) oxygen input, (2) sampling port, (3) UV-lamp, (4) quartz tube, (5) reaction vessel and (6) stirr bar.

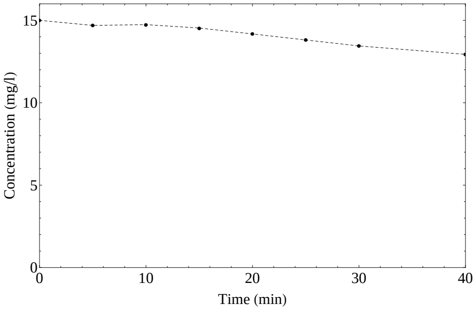
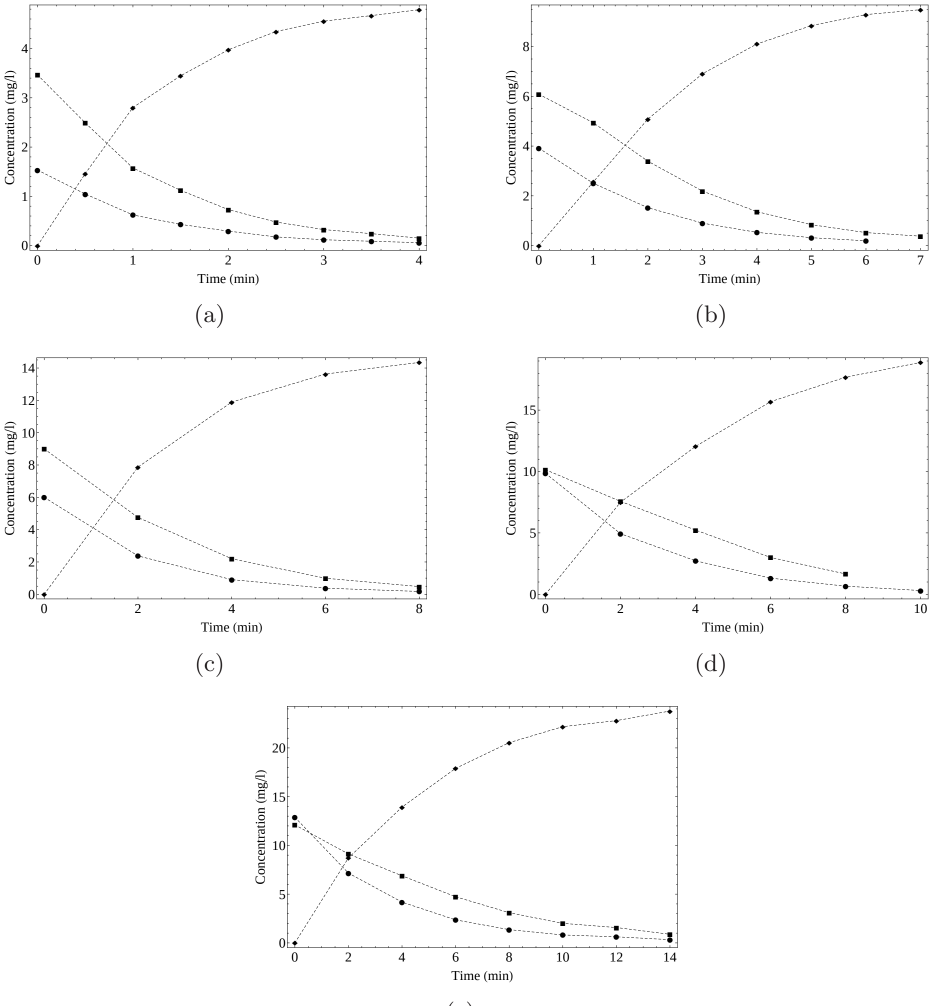


Figure 7.3: Concentration of MOX remaining in the solution for photolysis starting from a mixture with an initial concentration of 15 mg/l MOX, according to Van Doorslaer *et al.* [105].

the concentration of MOX remaining in the solution, adsorbed onto the catalyst and already degraded could be determined at every sampling time. The concentration of these different fractions in function of time, starting from a mixture with 15 mg/l MOX for photolysis and mixtures with 5, 10, 15, 20 and 25 mg/l MOX for photocatalysis, is shown in Figure 7.3 and Figures 7.4(a)–7.4(e), respectively.



(a)

(b)

(c)

(d)

(e)

Figure 7.4: Concentration of MOX remaining in the solution (●), adsorbed (■) and degraded (◆) for photocatalysis starting from a mixture with an initial concentration of (a) 5, (b) 10, (c) 15, (d) 20 and (e) 25 mg/l MOX, according to Van Doorslaer *et al.* [105].

7.3. Model development

In this chapter, a 2D stochastic CA-based model is used in which $\mathcal{T}^*$ is a square tessellation. Like in the previous chapter, the tessellation under consideration represents only a small part of the studied system of which the results will be translated to those for the original system making use of a scaling factor κ. The difference with the previous chapter, however, is that the phenomenon at hand is more complex and therefore demands a model with a more complex transition function. The latter also implies the need for more model parameters. Furthermore, as will be shown later on in this section, the use of a block CA is not favorable due to the model complexity. Since the block CA is not adhered to, boundary conditions need to be defined. In this case, as the tessellation is a small part of a larger whole, periodic boundary conditions are used, as such wrapping the tessellation on a torus and simulating the presence of an infinite/larger system.

Further, a set of three states is chosen as $S = \{\mathrm{H_2O}, \mathrm{MOX}, \mathrm{TiO_2}\}$. Naturally, there are more components present in the experimental reaction mixture, e.g. to buffer the solution. Nevertheless, all of these ingredients are non-reactive and in fact do not play a significant role in the process that is modeled. The two key components are the organic pollutant and the catalyst and for that reason, all non-reactive components are regarded as water.

To initialize the CA, every cell $c_{i,j} \in \mathcal{T}^*$ is assigned a state according to the proportion of the pollutant, catalyst and water in the batch reactor at the start of the experiment (see Section 7.2). This proportion can be based on the number of particles, the mass, the volume or the surface area of the three components in the reaction mixture. Although discussion on the exact mechanism behind photocatalysis is ongoing, it seems that particles have to be either in direct contact with the catalyst surface or in the close vicinity of this surface in order to degrade [101, 102, 103, 110], such that it seems appropriate to assign the initial condition of $\mathcal{T}^*$ according to the proportion of surface area. This leads to a surface area percentage of 8.3541×10^{-4}, 1.5465×10^{-2} and 99.9837 of MOX, $\mathrm{TiO_2}$ and $\mathrm{H_2O}$, respectively. It is brought to the attention of the reader that by making the aforementioned choice to assign the initial condition, that cells of $\mathcal{T}^*$ do not directly represent a molecule or not even a homogeneous amount of a component. Nevertheless, for the ease of reading, in what follows every cell is assumed to contain one ‘particle’.

In order to obtain a representative number of cells containing MOX and TiO_2, due to their low area percentage, a sufficiently large tessellation is required that nonetheless does not result in excessive computation times. A square tessellation with 4000^2 cells is chosen, yielding 134, 2474 and 1.5997×10^7 cells containing MOX, TiO_2 and H_2O, respectively, which are randomly distributed across the tessellation since the batch reactor is assumed to be well mixed (see Section 7.2). Furthermore, a scaling factor κ is defined that relates the initial number of cells in state MOX to the initial concentration of MOX in the batch reactor. Since the value of κ depends on the initial concentration of MOX in the reactor, it is different for each of the experimental data sets.

Note that only a small portion of the cells in the tessellation contains MOX or TiO_2, such that sparse matrices, only keeping track of the MOX and TiO_2 particles can be used to speed up calculations and spare computer memory. Further, a list that contains the number of MOX particles that is absorbed on every TiO_2 particle and a list that comprises the number of MOX particles that are free, adsorbed and degraded in every time step, are employed.

The transition function Ψ falls apart into two parts that describe reaction and diffusion [1], respectively.

7.3.1. Reaction

For the reaction part of Ψ, four stochastic key processes are considered, which are depicted in Figure 7.5 and are elaborated on in this paragraph.

Photolysis The first key process is photolysis, which is the direct degradation of MOX by UV light without assistance of the catalyst. In this case, only the cells containing MOX particles are updated, where every particle is broken down with probability P_{phot} (see Figure 7.6(a)).

---

[1] Throughout this chapter diffusion is employed to refer to the movement of molecules in the system, although in reality the driving force for movement in the system is advection.

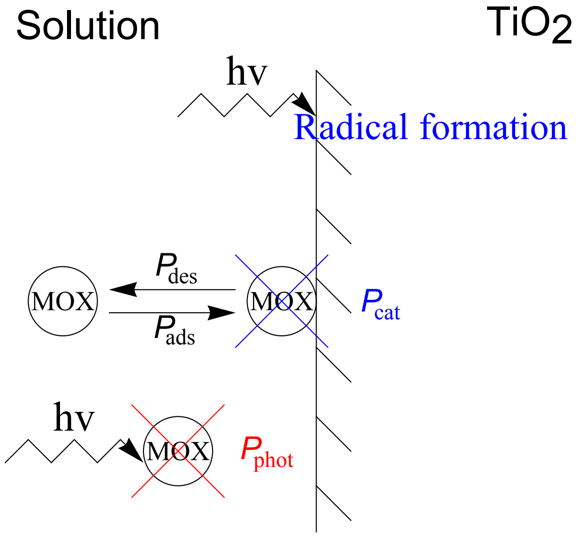


Figure 7.5: Schematic representation of the different components of the reaction part of Ψ.

Adsorption Adsorption of MOX particles onto the catalyst surface is the second key process [111]. To comply with the law of mass conservation, the cells that contain a TiO_2 particle are evaluated asynchronously in a random order (see Section 3.3), to prevent two different TiO_2 particles adsorbing the same MOX particle in the same time step. The reason why specifically random order is chosen here is because it has the simplest implementation and there is no real indication why every TiO_2 cell should be evaluated once before cells are evaluated a second time. All TiO_2 cells are equivalent and therefore the number of evaluations is more important than knowing which cells exactly are evaluated.

For this second key process, the neighborhood of the cells that contain a TiO_2 particle is checked and if a neighboring cell contains a MOX particle (if more are present, one particle is selected randomly), the MOX particle is adsorbed with probability P_{ads} (see Figure 7.6(b)). Here, it is assumed that every catalyst particle can adsorb an infinite number of MOX particles, which in practice will never occur due to the limited total number of MOX particles.

A neighborhood N for reaction still has to be defined, as well as the absolute time that corresponds to a discrete time step. On the one hand, a large neighborhood and a time step that corresponds to a long absolute time are preferable, because this would reduce the computation time. On the other hand, it has to be taken into account that the probabilities for reaction (P_{phot}, P_{ads}, P_{cat} and P_{des}) have to be comprised in $[0, 1]$, which poses a problem when opting for a very large neighborhood in combination with a large time step. Through preliminary model exploration, taking the above-mentioned constraints into account, a balance is found when employing the extended Moore neighborhood with radius two (see Figure 3.9) as the reaction neighborhood and with one time step equaling 0.1 s.

Catalysis MOX particles adsorbed on the catalyst surface can be degraded by photocatalysis, the third key process (see Figure 7.6(c)). Therefore, all cells containing catalyst particles are evaluated by randomly selecting one adsorbed MOX particle to be degraded with probability P_{cat}, even if more particles are adsorbed on the surface. As this degradation is catalyst assisted, P_{cat} is expected to be (much) larger than P_{phot}.

Desorption The fourth and last key process is desorption, where an adsorbed MOX particle can desorb again. The extended Moore neighborhood of a cell containing TiO_2 is checked for the presence of a cell containing H_2O (if there are more options, one cell is chosen at random), after which an adsorbed MOX particle can go back into solution with probability P_{des} (see Figure 7.6(d)). Again, all cells containing catalyst particles are evaluated asynchronously in a random order to respect the law of mass conservation.

7.3.2. Diffusion

The second part of the transition function Ψ, namely diffusion, is essentially deterministic, which means that particles diffuse with a probability of one in every time step. In the model implementation, only the MOX particles are allowed to diffuse, while the catalyst particles are static throughout the simulation. The diffusion of the MOX particles is done asynchronously in a random order, so that no two particles can diffuse to the same cell $c_{i,j}$ during a given time step.

A physical measurement of the speed of diffusion is the diffusion coefficient D which equals 7.13×10^{-5} cm^2/s for MOX under conditions similar to those

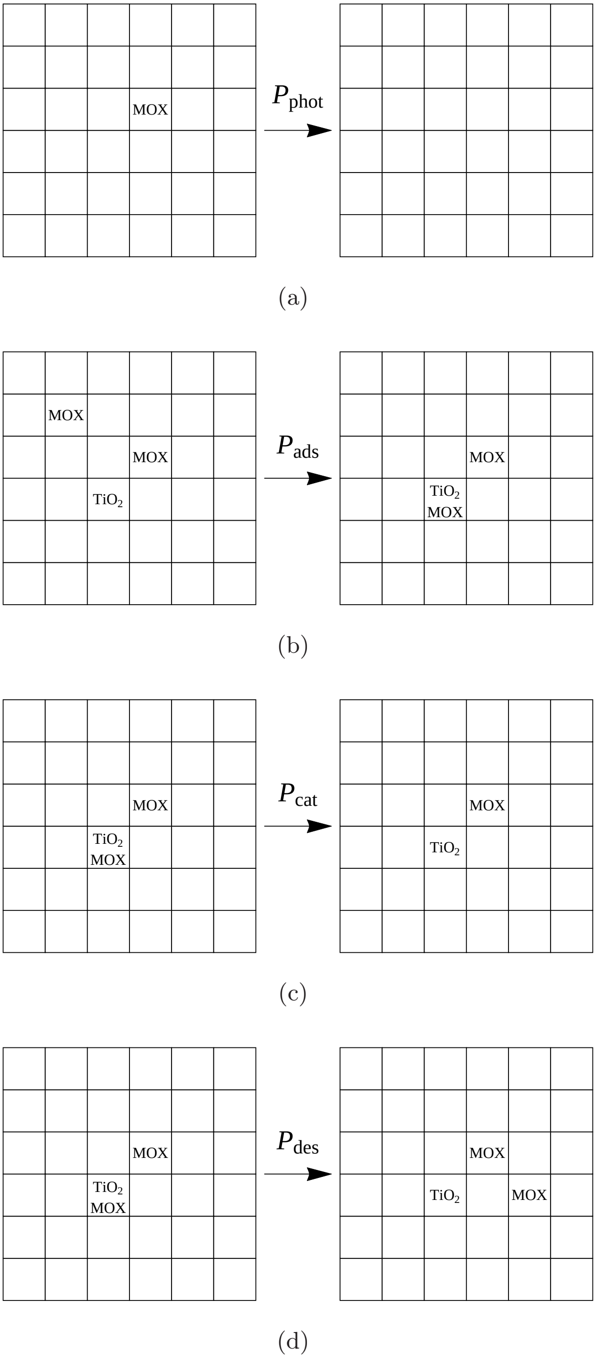


Figure 7.6: Reaction part of Ψ: (a) photolysis, (b) adsorption, (c) catalysis and (d) desorption.

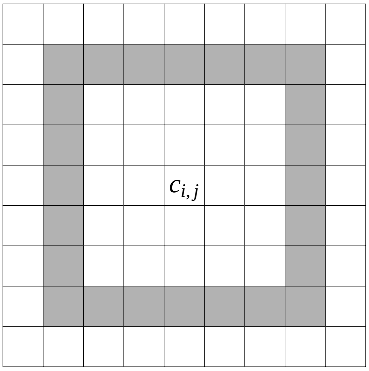


Figure 7.7: Neighborhood for diffusion (gray) of a cell $c_{i,j}$.

in the batch reactor [112]. The average distance a MOX particle travels in one time step therefore is $2\sqrt{D\Delta t}$ and for $\Delta t = 0.1$ s an average distance of 5.34×10^{-5} m per time step is obtained. Comparing this distance to the diameter of the batch reactor (i.e. 0.07 m, see Figure 7.2) and relating this ratio to the dimension of the tessellation, i.e. 4000 cells, results in an average diffusion distance of three cells per time step (see Eq. (7.2)). Figure 7.7 shows the neighborhood for diffusion employed in the CA-based model, where if cell $c_{i,j}$ contains a MOX particle, this particle will diffuse at random to one of the gray cells in its neighborhood at radius three.

$$\text{Diffusion distance} = \frac{2\sqrt{D\Delta t}}{0.07\ \text{m}}\sqrt{|\mathcal{T}^*|} = \frac{5.34 \times 10^{-5}\ \text{m}}{0.07\ \text{m}}\,4000\ \text{cells} \approx 3\ \text{cells}. \tag{7.2}$$

A flow chart of the developed CA-based model for the photocatalytic degradation of MOX is presented in Figure 7.8.

Finally, it is pointed out that in the current model implementation, the effect of pH and temperature are not taken into account, so that the focus can be put on the development of the CA-based model. The rate of photocatalytic mineralization, however, does not strongly depend upon pH for pH values ranging from four to ten [113] and is also not much affected by minor changes in temperature [103]. Given that the employed batch reactor is buffered and adiabatic (see Section 7.2), it may be argued that it is justified to leave aside both factors.

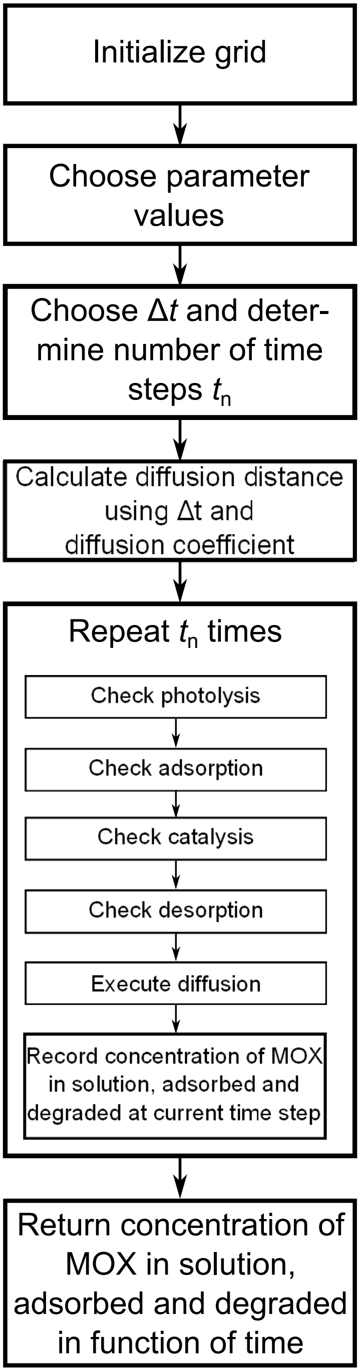


Figure 7.8: Flow chart of CA-based model for photocatalytic degradation of MOX.

7.4. Results and discussion

7.4.1. Verification

In order to check whether or not the CA-based model can be parameterized and which is the best way to do so, 50 time series of model outputs are generated through simulations for 500 s, an initial concentration of MOX of 10 mg/l and $P_{\text{phot}} = 3.5 \times 10^{-6}$, $P_{\text{ads}} = 0.6$, $P_{\text{cat}} = 0.01$ and $P_{\text{des}} = 0.08$, which are thereafter used as observed data. Such a repeated simulation is necessary to account for the stochasticity involved in the CA-based model. The choice of parameter setting is the consequence of a preliminary model exploration. Afterwards, the inverse problem is solved for each simulated model output, to verify whether or not the original parameter set can be retrieved.

Since there are four parameters to be optimized, a complete exhaustive search like was done in Chapter 6 is not feasible nor desirable. When parameterizing models that give rise to multiple local optima and where one model evaluation is highly time consuming, which is often the case for CA-based models, unnecessary model evaluations should be avoided and more protection against local optima is desired [114, 115]. Furthermore, when several computer nodes are at disposal for calculating several solutions simultaneously, iterative heuristics pose a problem, since a calculation is always based on the previous result. Further, the number of parameters to be determined is not excessive. Various optimization techniques were investigated in the framework of this dissertation, such as particle swarm optimization [116] and stochastic gradient descent [117]. In the end, a grid search, which evaluates the model in a number of predefined points in the parameter space Ω, was chosen as the preferred procedure given the aforementioned criteria.

For every set of model outputs that serves as observations, a grid search of 1600 Sobol generated parameter combinations in the search space is performed [118]. The optimal combination is chosen as the one that gives rise to the lowest weighted RMSE over the three time series of data (see Figures 7.4(a)–7.4(d)), resulting in 50 optimized values for every parameter, i.e. one for every series of observations. These values can be visualized pairwisely in two-dimensional plots. The result for P_{cat} and P_{ads} and for

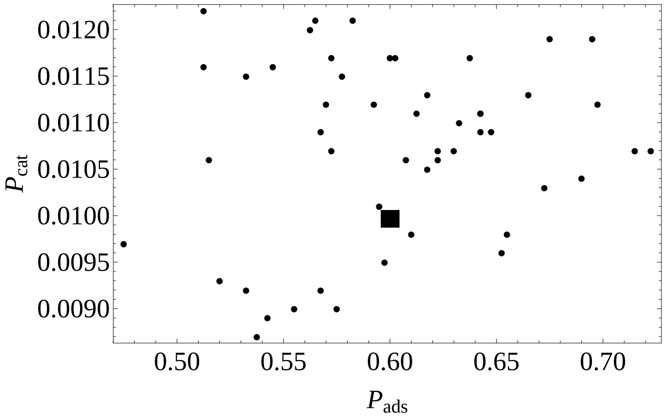


Figure 7.9: Values for P_{cat} and P_{ads} found through grid search (●) and the original parameter combination (▪) from which the observed data set for parameterization was obtained.

P_{phot} and P_{des} is shown in Figures 7.9 and 7.10, respectively. It can be observed that the parameter values are situated near the original value of the parameters. Due to the stochasticity involved and the fact that only a limited number of combinations of parameter values in the search space is investigated, it was to be expected that the exact values would not be found. The plots for other parameter combinations yield similar results and therefore, it may be concluded that the parameters corresponding to a given set of observed data can be identified by means of a grid search.

7.4.2. Sensitivity analysis

As a part of the model exploration, but also to get an idea on how much effort should be spent on each parameter during the parameterization, an SA is performed. Like explained in Section 4.4, a screening of the parameters is performed based on the EE method. The simulated outcome Y on which the influence of a small perturbation of the parameter values is studied, is the weighted RMSE between the simulated and experimental time series for an initial concentration of MOX of 10 mg/l. The simulated time series of data are obtained through 4200 time steps of the CA-based model, which is the number of time steps corresponding to the 7 min during which the samples were taken in the photocatalytic experiment with initial concentration of MOX of 10 mg/l and taking into account $\Delta t = 0.1$.

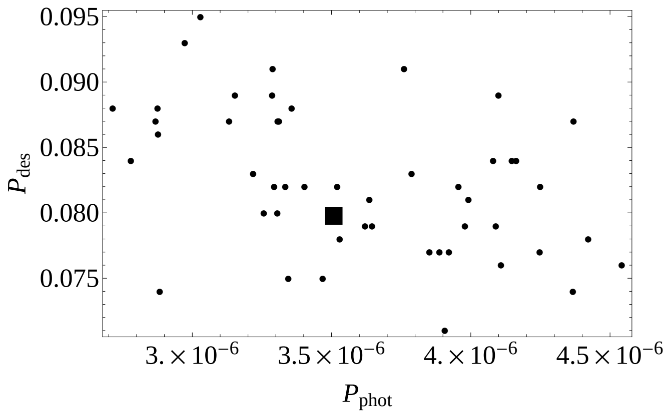


Figure 7.10: Values for P_{phot} and P_{des} found through grid search (●) and the original parameter combination (▪) from which the observed data set for parameterization was obtained.

Further, 50 trajectories, i.e. $\mathcal{R} = 50$, are used to calculate the EE with the starting points of the trajectories chosen by means of a Sobol sequence [118] in the search space Ω. The perturbation is achieved by multiplying the parameter values at the starting points of the trajectories by 1.1, so that the perturbation for each parameter is, relatively seen, equal. Figure 7.11 shows the sensitivity measure H for the four model parameters. It can be observed that none of the parameters is redundant and that small variations of P_{phot} and P_{des} have less pronounced effects than small variations of P_{cat} and P_{ads}, so that a correct determination of the latter two parameters is more important than the former two. Although obtained for an initial concentration of MOX of 10 mg/l, the results in Figure 7.11 are also valid for the other initial concentrations.

7.4.3. Parameterization

The inverse problem of retrieving the values of the four model parameters corresponding to the observed data from Section 7.2, is solved in two steps. In the first step, P_{phot} is determined using the data of photolysis, i.e. the degradation of MOX through UV light in absence of a catalyst ($P_{\mathrm{ads}} = P_{\mathrm{des}} = P_{\mathrm{cat}} = 0$). Since an approximative value of P_{phot} suffices and in this case there is only one parameter to be determined, it is opted to perform an exhaustive search for the values of P_{phot} with a limited number of decimal places.

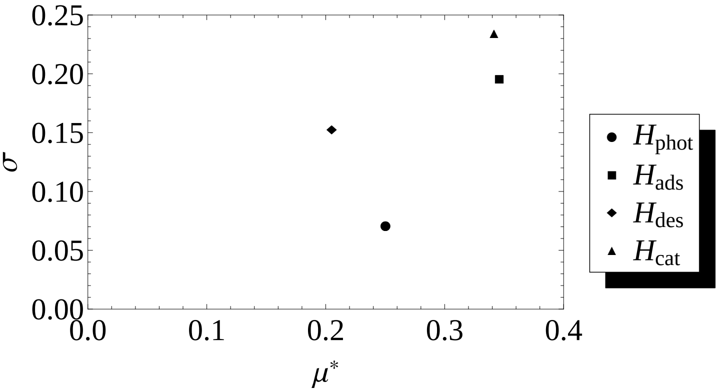


Figure 7.11: Screening of the model parameters based on the EE method.

The observed data for photolysis are obtained from a reaction mixture with an initial concentration of MOX of 15 mg/l that was sampled during 40 min (see Figure 7.3), which are translated to an initial number of MOX particles present in $\mathcal{T}^*$ and a number of time steps (using $\Delta t = 0.1$ s), respectively. At every time step during the evolution of the CA-based model, the simulated number of free, adsorbed and degraded MOX particles is registered. These numbers, together with the scaling factor κ mentioned in Section 7.3 and the relationship between a discrete time step of the model and absolute time, resulted in the simulated concentration of the different fractions of MOX as a function of time. As for the verification of the model earlier on in this section, the RMSE is used as measure of fitness. The parameterization is repeated ten times and the average of these repetitions is used to obtain the optimized value for P_{phot}, i.e. the value of P_{phot} which results in the lowest RMSE. The found optimized value is 6.41×10^{-5}.

The probabilities of the four key processes (see Figure 7.5) are assumed to be independent of the initial concentration of MOX or TiO_2. This allows to employ the formerly found value of P_{phot} for the parametrization of the model with experimental data from photocatalytic degradation, as such reducing the number of parameters that have to be determined through a grid search from four to three. There is, however, an increase in turbidity in the lab-scale reactor by adding the catalyst powder, so that the light scattering increases and P_{phot} decreases. Nevertheless, the choice is made not to include P_{phot} in the grid search optimization as it is much smaller in comparison to the three other model parameters (see infra) so that a small alteration of P_{phot} does not seriously affect the final simulated outcome for photocatalysis.

For the determination of P_{ads}, P_{des} and P_{cat}, there are five different data sets (see Section 7.2, one data set for each initial concentration) that can be used, assuming that the three parameters are independent of the initial concentration of MOX. In theory, the three parameters could therefore be determined using the data from one initial concentration of MOX as was the case for the assessment of P_{phot}. However, from the SA it was concluded that a correct determination of especially P_{ads} and P_{cat} is important, as these parameters have a more pronounced influence on the model outcome than the other parameters. For this reason, all five data sets are used for the parameterization of the CA-based model in order to account for variations in set-up and measurement. Following preliminary data exploration, a domain for grid search was chosen as given in Table 7.1, with L, U being respectively the lower and upper bound of the search interval per parameter and V the step size within that search interval.

Table 7.1: Set up for grid search, with L, U being respectively the lower and upper bound of the search interval for the parameter and V the step size within that search interval.

	L	U	V
P_{ads}	0.5	0.86	0.04
P_{des}	0.01	0.09	0.01
P_{cat}	0.001	0.0105	0.0005

The RMSE, resulting from comparing the experimental data from Section 7.2 to the simulated output of the different fractions of MOX, is calculated for all parameter combinations that are possible according to Table 7.1. To determine the parameter combination that gives rise to model simulations that best correspond to the experimental data, a graphical representation is employed where a separate graph is used for each of the three parameters.

Figure 7.12 shows the result for P_{des}, where for each initial concentration of MOX from the experimental data the fitness of each discrete point of the parameter space is depicted, with larger circles corresponding to greater fitness. This fitness is obtained by taking the lowest RMSE found through grid search of all the parameter combinations that contain that specific

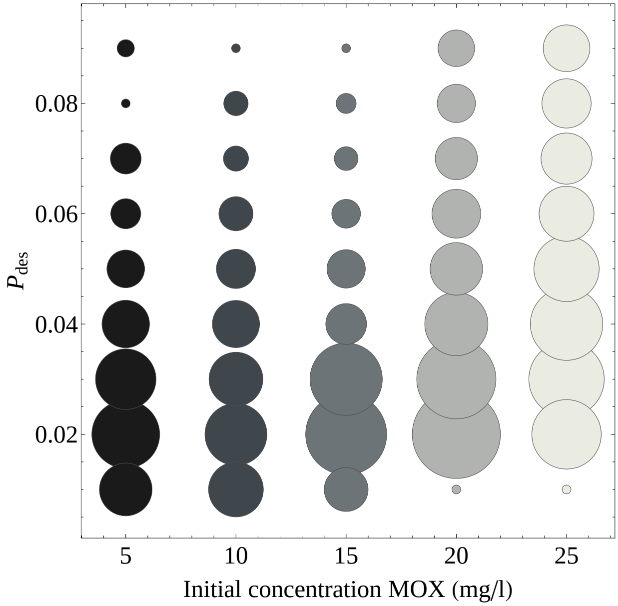


Figure 7.12: Graphical representation for the determination of P_{des}, where larger circles correspond to a greater fitness.

discrete parameter value. Figure 7.12 confirms the assumption that the probabilities of the key processes are independent of the initial concentration of MOX. As can be seen, $P_{\mathrm{des}} = 0.02$ gives the best result considering all different initial concentrations of MOX. The same procedure was followed for the two remaining parameters, resulting in $P_{\mathrm{ads}} = 0.78$ (see Figure 7.13) and $P_{\mathrm{cat}} = 0.0015$ (see Figure 7.14).

Figure 7.15 shows the comparison of the model simulation with the fitted values for P_{ads}, P_{des}, P_{phot} and P_{cat} and the experimental data for a photocatalytic degradation of a mixture with an initial MOX concentration of 10 mg/l, in which the diamonds represent the fraction of MOX still in solution, the circles represent the fraction of MOX broken down, the dashed lines represent the experimental data and the dotdashed lines represent the simulated data. The adsorbed fraction is not represented in the figure in order not to overcomplicate the figure. This figure shows that the simulated outcomes approximate the experimental ones quite good. Better results for a single concentration are most probably attainable. However, the goal was to find a parameter set that gives a good fit for each of the five studied concentrations of MOX and not a parameter set for each individual concentration.

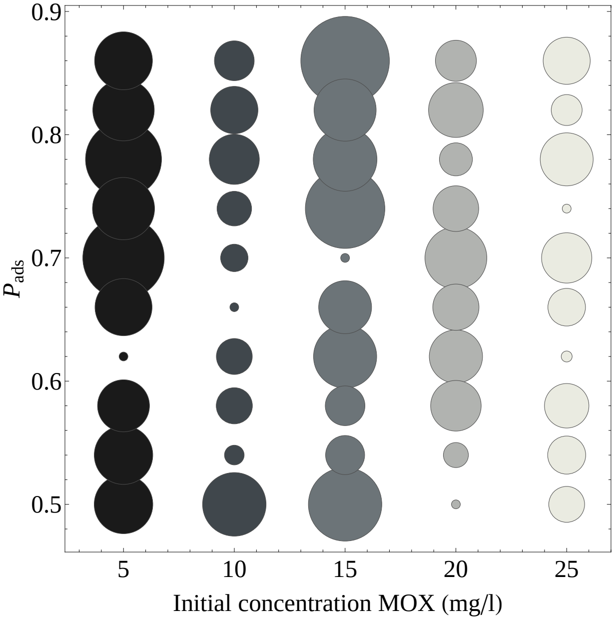


Figure 7.13: Graphical representation for the determination of P_{ads}, where larger circles correspond to a greater fitness.

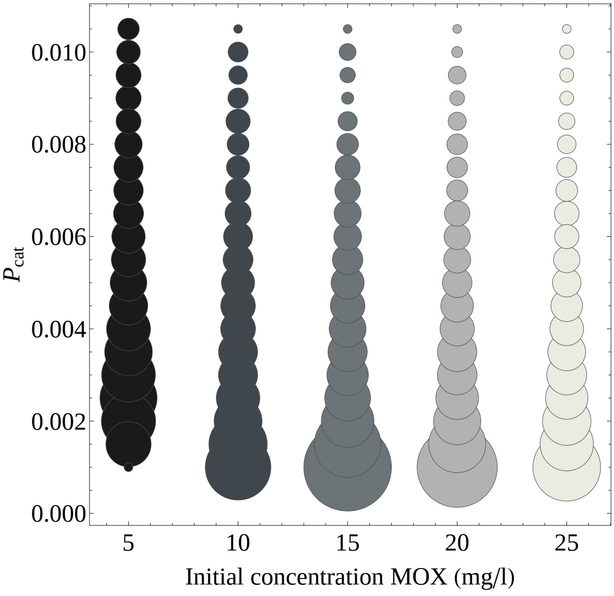


Figure 7.14: Graphical representation for the determination of P_{cat}, where larger circles correspond to a greater fitness.

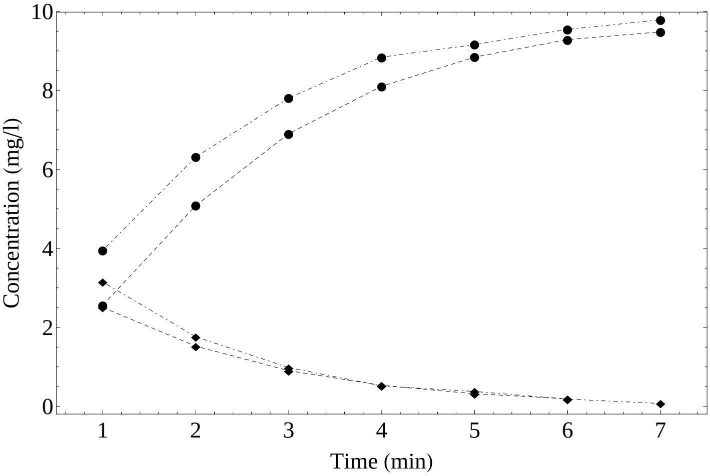


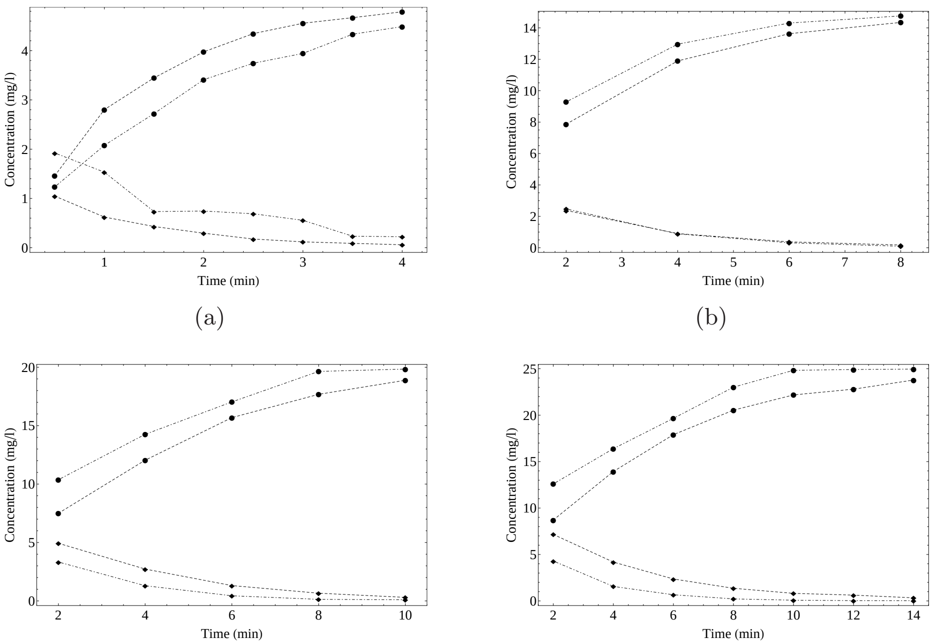
Figure 7.15: Concentration of MOX remaining in the solution (●) and degraded (◆) for photocatalysis starting from a mixture with an initial concentration of 10 mg/l MOX. Dashed lines (experimental data) and dotdashed lines (simulated data) are drawn to guide the eye.

A similar result is obtained for the other initial concentrations (see Figures 7.16(a)–7.16(d)).

7.4.4. Hospital wastewater treatment

In order to illustrate the potential of the developed model, the remainder of this section deals with an exemplary case study for the photocatalytic degradation of MOX.

Companies and institutes are fined if their wastewater effluent exceeds legal concentration limits for harmful substances. As a result, many companies and institutes have their own wastewater treatment plant in order to comply with regulations. However, FQs are not completely removed by traditional treatment (see Section 7.1) and can therefore pose a significant problem when the use of FQs is ample, as is the case at the accident and emergency department of hospitals, where average CIP concentrations of 60 μg/l are measured [119]. It is assumed here that as the successor of CIP, MOX will eventually reach similar concentrations in the future. Instead of treating the relatively large volume of wastewater coming from the whole hospital with photocatalysis after the traditional treatment, it would be more energy and time efficient to treat the effluent from first aid, rich in



(a) (b)

(c) (d)

Figure 7.16: Concentration of MOX remaining in the solution (●) and degraded (◆) for photocatalysis starting from a mixture with an initial concentration of (a) 5 mg/l MOX, (b) 15 mg/l MOX, (c) 20 mg/l MOX and (d) 25 mg/l MOX. Dashed lines (experimental data) and dotdashed lines (simulated data) are drawn to guide the eye.

FQs, separately before aggregating it with the rest of the hospital water. Namely, TiO_2-based systems for treating large volumes of contaminated water are suffering from low yield and consequently their economic feasibility is questioned [104].

It is presumed that a test phase reactor to treat wastewater rich in FQs, used in the first aid of a hospital, is an upscaled version of the lab-scale batch reactor in Figure 7.2 with a volume of 7.5 l and a reaction time of the effluent of 30 min. Further, it is assumed that in the future a maximum concentration of MOX in effluent of 5 μg/l will be allowed. The parameterized CA-based model is used to search an adequate catalyst concentration for the aforementioned assumptions. A high concentration of photocatalyst speeds up degradation, but an excess must be avoided to ensure total absorption of efficient photons [103, 120, 121]. Namely, an unfavorable light scattering and reduction of light penetration into the solution is observed with excess photocatalyst loading. Thus, the minimal catalyst concentration so that the maximum allowed concentration of MOX is obtained after photocatalysis, is desired. Since only one parameter has to be determined, this was done through an exhaustive search as was done for the retrieval of P_{phot} for photolysis earlier on. Figure 7.17 shows the difference between the simulated remaining amount of MOX in the effluent under the conditions mentioned above and the maximum allowed concentration of MOX in the effluent in function of the mass of catalyst added to the reactor. To obtain adequate usage of the catalyst, the maximum allowed and simulated amount of MOX should be equal, which resulted in 0.165 mg of TiO_2 to be added to the reactor.

7.5. Conclusions

The developed CA-based model with the incorporation of the four key processes, being photolysis, adsorption, desorption and photocatalysis, enables the simulation of the photocatalytic degradation of FQs. It can be argued that the incorporation of photolysis is redundant as the impact of this process on the simulated model outcome is small in comparison to the other three key processes. However, it was decided to keep photolysis in the model, because it does have an influence, it is a real physical process that takes place and its incorporation does not pose a large burden on the com-

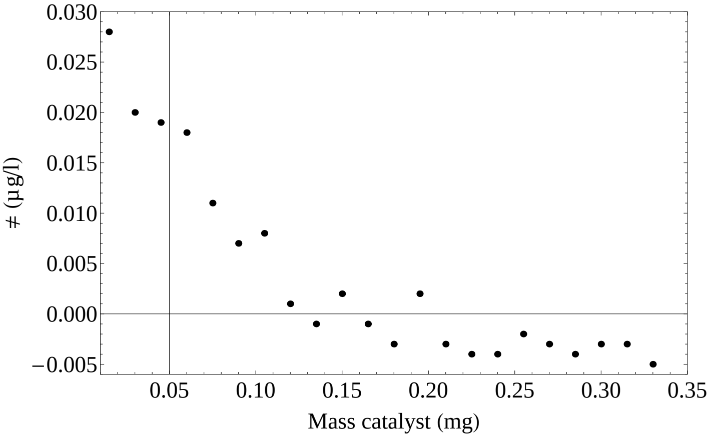


Figure 7.17: Difference between the simulated remaining amount of MOX in the effluent and the maximum allowed concentration of MOX in the effluent in function of the mass of catalyst (mg) added to the reactor.

putation times.

The results obtained with the sensitivity analysis show that small perturbations of P_{cat} and P_{ads} have a more pronounced effect on the model outcome than the two other parameters, implying that their correct determination is extra important. The parameterization of the CA-based model was done via grid search using experimental data from a lab-scale batch reactor and a potential application of the CA-based model was illustrated by means of a case study.

Like in the previous chapter, the system being modeled is well mixed such that the choice is made to model only a part of the entire system which is then afterwards scaled in order to minimize computation times. However, in contrast with Chapter 6, the block CA paradigm is not adhered to here. The use of sparse arrays in this case is preferred as it allows to keep track of the limited number of fixed TiO_2 cells and moving MOX cells among the vast number of H_2O cells. Employing a block CA in this case would result in a scanning of the entire tessellation after every diffusion step in order to retrieve the small number of cells that are of interest, which would result in a severe computational effort. The choice not to employ a block CA calls for boundary conditions. Here, periodic boundary conditions are used since the tessellation represents a small part of a larger system and because of their ease of implementation.

Having made the aforementioned choices in order to reduce computation times, the model is sufficiently fast such that a preliminary search of the parameter space was possible. In this way, smaller bounds for each parameter could be found. Regarding the parameterization, here all different data sets were used to find one single combination of parameter values, since they are assumed to be independent of the initial concentration of MOX. Finally, the effect of diffusion in this case study is not incorporated into the other parameter(s), but modeled explicitly. From this chapter, it is clear that there is a relationship between the physical dimension of a tessellation cell, the physical time that corresponds to a discrete time step and the diffusion coefficient. When, like was the case here, the diffusion coefficient is known, choosing a number of seconds that correponds to a time step at the same time determines the physical size of a cell and vice versa.

8 Oil migration

Oil migration is an important process in the formation of fat bloom on chocolate-coated confectionery, leading to consumer rejection. However, the exact mechanisms behind this phenomenon are still not completely elucidated, which hampers the development of a mathematical simulation model. In this chapter, a stochastic CA-based model is proposed and parameterized using experimental data obtained from confectionery model systems. This CA-based model is shown to be able to describe the oil migration in an adequate manner and can therefore be used to calculate an effective diffusion coefficient. Further, the potential of a CA-based approach for the further investigation of the fat bloom mechanisms is demonstrated by means of a case study where capillary rise is incorporated in the model.

8.1. The phenomenon

Belgian pralines, commonly known as Belgian chocolates, were first introduced in 1912 by Jean Neuhaus II, a Belgian chocolatier. They usually contain a hard chocolate coating with a softer, for example (hazel)nut-based, filling. Today, Belgian pralines still have an excellent reputation on the international market. However, the whitish haze formed over time on the surface of chocolate, known as fat bloom (see Figures 8.1(a) and 8.1(b)), poses a worrisome problem hampering the export of these products [122]. This haze is the result of dispersion of light on small fat crystals that are formed when recrystallization occurs at the surface. Figure 8.2 shows a microscopic image (156× magnification) of the surface of a bloomed praline. Fat bloom occurs on all chocolate products, but the presence of a liquid filling accelerates the process as the filling oils are often completely liquid at room temperature and can therefore easily transfer through the chocolate coating to the surface [123].

Bloomed pralines are harmless and still consumable, but a softening of the chocolate coating, a hardening of the filling, a flattening of the taste and most importantly a rejection of the pralines by the consumers due to the

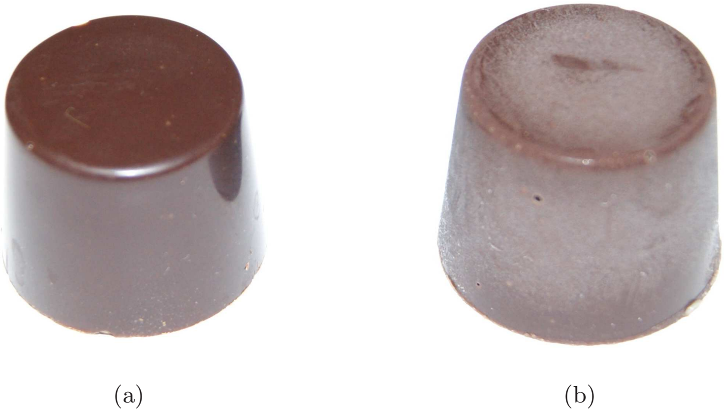

(a) (b)

Figure 8.1: Chocolate praline: (a) no fat bloom and (b) with presence of fat bloom.

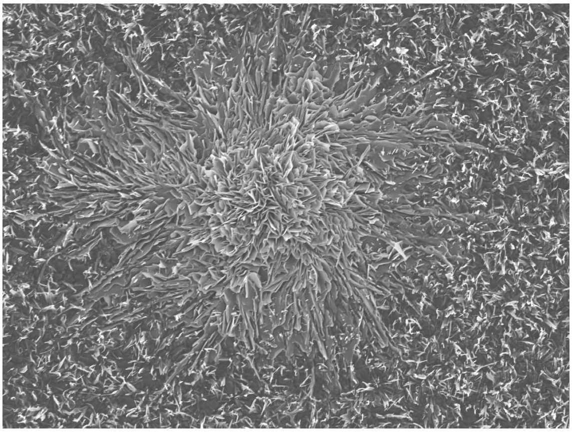


Figure 8.2: Microscopic image of the surface of a fat bloomed praline (156× magnification).

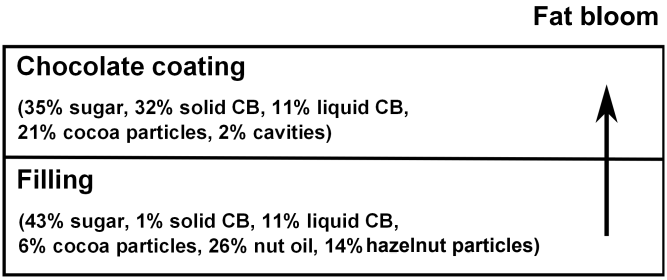


Figure 8.3: Schematic representation of fat bloom in chocolate confectionery, with indication of the composition of the different layers in volume percentages.

association with inferior and expired products may occur [124]. A solution to this problem would therefore be very valuable for this multimillion dollar industry. Unfortunately, the actual mechanisms behind fat bloom remain speculative and a more thorough understanding is necessary to better abate quality deterioration. This knowledge can aid a manufacturer to determine *a priori* the effect on the rate and amount of fat bloom when changing an ingredient or process. Therefore, there is a need to develop better models that combine mass transfer with the phase behavior for accurately predicting the migration of liquid fat and the occurrence of fat bloom [41, 125, 126].

The phenomenon of fat bloom is presented schematically in Figure 8.3. A crucial step in the formation of fat bloom is the migration of liquid fat to the surface. Several hypotheses and mechanisms have been put forward to explain oil migration in chocolate. Originally, the driving force for the migration was believed to be diffusion due to a difference in liquid fat content [41], but nowadays diffusion is ascribed to a gradient in triacylglycerol (TAG) concentration between the chocolate coating and the filled center [125]. The addition of extra cocoa particles and sugar particles is said to retard the oil migration rate, because they are impenetrable to oil [41]. However, a recent study has shown that cocoa particles disrupt the formation of the cocoa butter (CB) crystal network such that the resulting crystal network is less dense and more permeable to oil [127]. Besides diffusion, capillary forces may play a significant role and have therefore been proposed as an alternate mechanism of oil migration [123, 125].

Another hypothesis for oil migration states that the increase in volume when CB melts forces liquid fat to the surface through pores and micro fractures formed during crystallization [128]. Several other theories focus on the thermal stability of the different polymorphic forms of CB as the cause of fat bloom [125], but although bloom formation is accompanied by a polymorphic transition of CB, it is by itself not sufficient to always cause visual fat bloom [123]. In literature, there is neither a consensus on which mechanisms actually play a role nor on their relative importance, but most papers focus on the diffusion of TAGs and the capillary rise or a combination of both.

8.2. Data acquisition

In order to collect the necessary experimental data to calibrate the CA-based model, a number of confectionery model systems were produced and analyzed. A standard dark chocolate with 25% (m/m) fat (Barry Callebaut, Wieze) was adjusted to 35% fat (m/m) by adding CB. In order to monitor oil migration in this set-up, a hazelnut filling was produced by Barry Callebaut (Wieze, Belgium). The hazelnut base was prepared by mixing 50% (m/m) sugar, with 50% (m/m) hazelnuts. A standard dark chocolate was added in a ratio of 25/75 chocolate/hazelnut filling. Confectionery model systems were produced in small cylinders having a height of 4 cm and a radius of 1.3 cm by adding a 0.5 cm chocolate layer on top of a 3.5 cm filling layer (see Figure 8.4). The different samples were stored at 20 °C and monitored for 48 weeks (i.e. 29030400 s) by analyzing three samples on a two- to three-weekly basis.

By using a LKB Bromma 2218 Historange microtome, ten slices of 0.5 mm were obtained from the original chocolate layer. Layers two, four, six and eight (with layer one being the layer in contact with the filling) were further analyzed in order to monitor the migration of filling oil through the chocolate. Firstly, the fat phase from every layer was extracted by dissolution of the fat. The chocolate sample was dissolved in 10 ml of petroleum ether, vigorously shaken and centrifuged (10 min, 3000 rpm), after which the solvent of the supernatant was evaporated at 50 °C. The remaining fat was dissolved in the mobile phase, being acetonitrile/dichloromethane 70/30 (v/v). In order to assess the degree of migration of the filling oil into the chocolate during the storage period, the TAG composition of the different layers, obtained through HPLC of the fat phase of every layer,

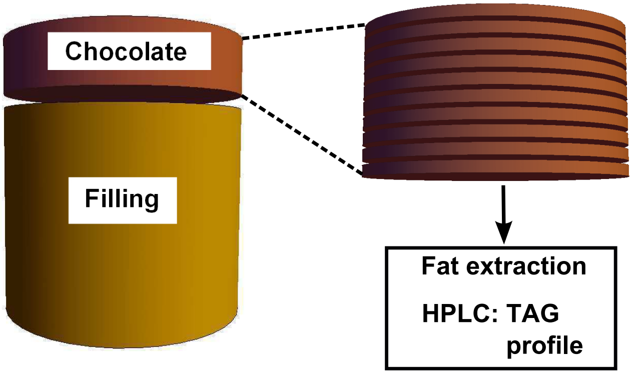


Figure 8.4: Data acquisition: On the left the chocolate model system and on the right the sliced chocolate layer and subsequent analysis steps.

was analyzed. As shown in [129], a number of these TAGs could be used as markers for the migration of fat through the chocolate.

Figures 8.5 and 8.6 show the mass of nut oil and liquid CB, respectively, in the studied layers through time. It is important to point out that the measurements at every sampling time constitute an average of three different confectionery model systems and that the measurements for the different layers at one sampling time originate from the same set of confectionery model systems. However, as the measurement method is destructive, the measurements at different sampling times originate from a different set of three confectionery model systems. Further and more detailed information on the data acquisition can be found in [129].

8.3. Model development

In this section, a stochastic, homogeneous 2D CA-based model describing oil migration, a key process in fat bloom, is proposed.

A square tessellation $\mathcal{T}^*$ is used with periodic boundary conditions along the horizontal axis and fixed boundary conditions along the vertical axis. A schematic representation of the tessellation $\mathcal{T}^*$ is shown in Figure 8.7. The fixed boundaries are dictated by the experimental confectionery model systems and mean that through these boundaries, no movement is possible. On the other hand, the periodic boundaries in the horizontal direction

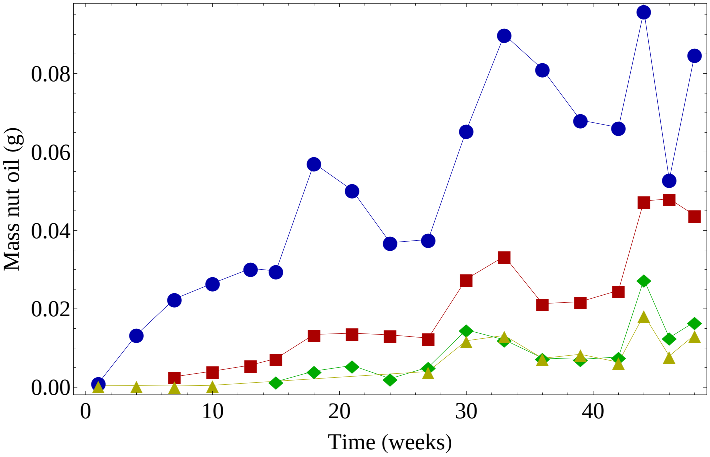


Figure 8.5: Evolution of the mass of nut oil in function of time for layers two (blue, ●), four (red, ■), six (green, ◆) and eight (yellow, ▲).

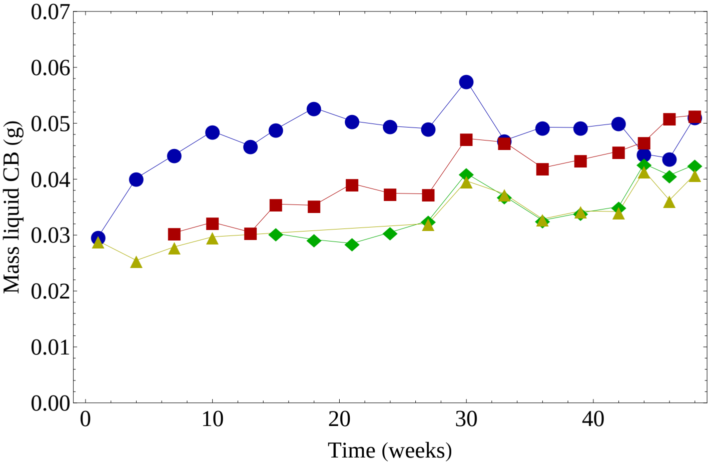


Figure 8.6: Evolution of the mass of liquid CB fat in function of time for layers two (blue, ●), four (red, ■), six (green, ◆) and eight (yellow, ▲).

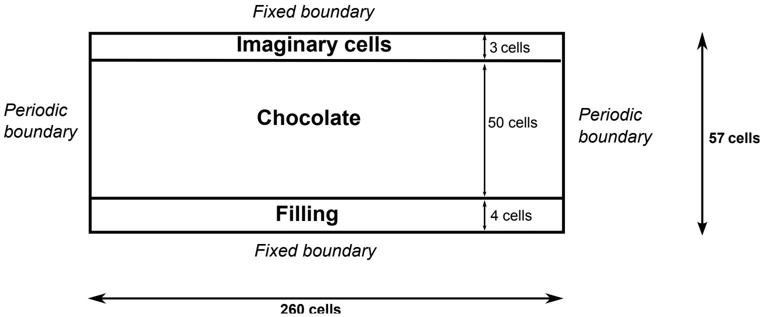


Figure 8.7: Schematic representation of the square grid.

make that every cell has the same number of neighbors according to the horizontal axis which facilitates calculations. Here, the first and the last column are considered as adjacent as such simulating a continuous space according to the horizontal axis.

The tessellation $\mathcal{T}^*$ as a whole represents of a vertical cross-section of the system wherein oil migration is studied. This is different from the case studies in the two previous chapters, where the tessellation represented a part of a larger whole that had to be scaled up afterwards. In this case, the tessellation captures the entire system that is being studied, although only in two dimensions. By presuming the side length of a square cell, Δx, to be 0.0001 m, 50 rows and 260 columns of cells are needed to represent a chocolate layer of height 0.5 cm and diameter of 2.6 cm. The value of Δx is the result of a trade-off. On the one hand, setting Δx as large as possible is desired to reduce the computation time of the simulations. On the other hand, a small enough grid resolution and therefore small enough Δx is wanted to prevent overly sensitive model parameters that result in very fluctuating solutions from simulation to simulation. The used value of Δx keeps the middle between the aforementioned requirements.

Besides the chocolate layer, $\mathcal{T}^*$ encloses two more parts being three rows of imaginary cells on top of the chocolate layer for imposing the fixed boundary conditions and four rows of filling below it. The filling is considered as an infinite source of nut oil which is replenished at every time step by setting the states of these filling cells back to their state at $t = 0$ after every time step.

Further, seven different discrete states are discerned, i.e. $S = \{$cavity, solid CB, liquid CB, sugar, cocoa particle, nut oil, hazelnut particle$\}$, where a

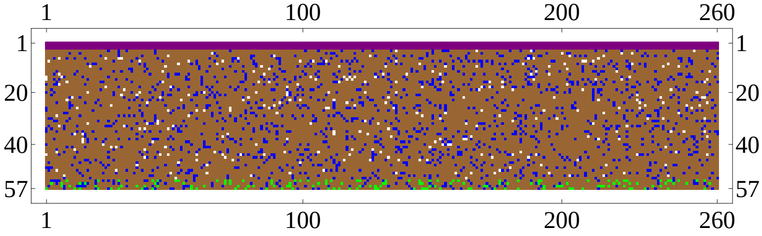


Figure 8.8: Initial condition of the CA, with cavities in white, liquid CB in blue, nut oil in green, imaginary cells in purple and the other states in brown.

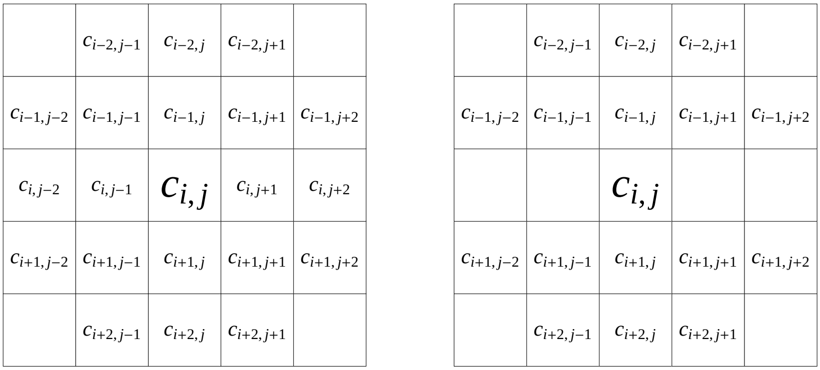
cell in a certain state represents a homogeneous amount of an ingredient, which is a simple approximation intended to capture the synoptic effects of oil migration [76, 77]. Diffusion is the consequence of a gradient in TAGs (see Section 8.1). However, here nut oil is considered as a general state rather than a different state for each TAG in order not to overcomplicate the model.

The initial configuration of the CA is established by assigning every $c_{i,j} \in \mathcal{T}^*$ for both the chocolate coating part and the filling a state according to the volume proportion of the different states in S as depicted in Figure 8.3. The volume percentages are chosen here since the presence of cavities in the chocolate coating can only be expressed on a volume base. For the basic model, all initial states are randomly distributed across $\mathcal{T}^*$. Further, a scaling factor κ is determined. This factor is used to convert the simulation results that are expressed as a number of cells into their corresponding magnitude in grams by taking into account the total mass in grams of the liquid fat components at $t = 0$ and the number of cells in $\mathcal{T}^*$ at $t = 0$ used to represent this amount. Figure 8.8 shows the initial condition of the CA where the cavities are colored white, liquid CB blue, nut oil green, imaginary cells purple and for reasons of clarity, the other states comprised in S are colored brown.

The transition function Ψ consists of two parts that describe dissolution of solid CB, and oil migration, respectively.

8.3.1. Dissolution

The first part of the transition function describes the transition of solid CB to liquid CB. Under normal conditions there exists a chemical equilibrium



	$c_{i-2,j-1}$	$c_{i-2,j}$	$c_{i-2,j+1}$	
$c_{i-1,j-2}$	$c_{i-1,j-1}$	$c_{i-1,j}$	$c_{i-1,j+1}$	$c_{i-1,j+2}$
$c_{i,j-2}$	$c_{i,j-1}$	$c_{i,j}$	$c_{i,j+1}$	$c_{i,j+2}$
$c_{i+1,j-2}$	$c_{i+1,j-1}$	$c_{i+1,j}$	$c_{i+1,j+1}$	$c_{i+1,j+2}$
	$c_{i+2,j-1}$	$c_{i+2,j}$	$c_{i+2,j+1}$	

(a)

	$c_{i-2,j-1}$	$c_{i-2,j}$	$c_{i-2,j+1}$	
$c_{i-1,j-2}$	$c_{i-1,j-1}$	$c_{i-1,j}$	$c_{i-1,j+1}$	$c_{i-1,j+2}$
		$c_{i,j}$		
$c_{i+1,j-2}$	$c_{i+1,j-1}$	$c_{i+1,j}$	$c_{i+1,j+1}$	$c_{i+1,j+2}$
	$c_{i+2,j-1}$	$c_{i+2,j}$	$c_{i+2,j+1}$	

(b)

Figure 8.9: Neighborhoods in a square grid for (a) transition and (b) migration.

between the amount of liquid CB and the amount of solid CB in a certain region of the chocolate coating, which is described by Eq. (8.1) [130]. However, as nut oil migrates through the chocolate layer, this equilibrium is disturbed because nut oil acts as a solvent for solid CB. This results in an increase of the amount of liquid CB in the chocolate layer. Further, from the experimental data (see Figures 8.5 and 8.6) a maximal amount of liquid CB per layer of 0.05 g, obtained when a layer is saturated with nut oil, was deduced. Here it is assumed that the increase in liquid CB in a layer is only caused by the transition of solid CB. The latter means that the increase (or decrease) in liquid CB in a certain layer is not due to migration of liquid CB through the layers (see infra).

$$A \text{ (liquid CB)} \underset{\beta}{\overset{\beta_{-1}}{\rightleftharpoons}} B \text{ (crystalline CB)}. \tag{8.1}$$

To model this transition reaction, a model parameter P_t is introduced, which represents the transition probability of solid CB to liquid CB. This probability P_t is multiplied with the number of cells in the nut oil state in the transition neighborhood N_t, which is depicted in Figure 8.9(a), to obtain a dissolution probability P_s. This is done to account for the fact that a higher concentration of nut oil results in a higher probability of dissolution. However, if the calculated value for P_s exceeds 1, P_s is set equal to 1 to ensure that $P_s \in [0, 1]$. Further, when the maximal amount of liquid CB in that layer is reached, P_s is set to 0.

8.3.2. Migration

For the migration part of the transition function Ψ it is assumed that only nut oil migrates through the chocolate coating. This choice is based on preliminary results with an earlier developed model where both nut oil and liquid CB were allowed to migrate. However, by doing this, the experimental data in Figures 8.5 and 8.6 could not be reproduced. Both types of liquid fat clearly have different properties and migration potential, with the nut oil being more mobile than the liquid CB. Therefore, only cells in the nut oil state are evaluated for this part of the model evaluation. The diffusion is governed by a diffusion probability P_d and proceeds in the direction of the local gradient in concentration. To incorporate this gradient-driven diffusion, a neighborhood for diffusion N_d is introduced in Figure 8.9(b). This neighborhood is split up into eight cells forming the upper part (cells closer to the open air) and eight cells forming the lower part (cells closer to the filling).

The number of cells in the nut oil state are counted in both the upper and lower part of N_d and it is presumed that nut oil will move one cell according to the vertical gradient, i.e. in the direction having the smallest number of cells in this state. If both the upper and lower part of N_d contain the same number of cells in the liquid state, the liquid fat in $c_{i,j}$ will move one cell with probability P_d (to the left or to the right, chosen at random). Clearly, this entails that liquid fat tries to move in every time step. In the CA-based model, the motion dynamics of nut oil are simulated by switching the state of the central cell $c_{i,j}$ and the state of the cell it moves towards. Since the diffusion coefficient of nut oil in chocolate confectionery is not known, Δt, i.e. the number of seconds corresponding to one discrete time step of the CA-based model, is set as a model parameter. The total number of time steps of the CA-based model is denoted t_n and is given by Eq. (8.2):

$$t_n = \lfloor 29030400/\Delta t \rfloor . \tag{8.2}$$

Normally, diffusion cannot occur if the nut oil has to pass through sugar, because this compound is impermeable to oil (see Section 8.1). However, to account for the possibility of oil migration by capillarity, volume and thermal effects as well as to take the influence of cocoa particles on the crystalline structure (see Section 8.1) into consideration, ‘diffusion through sugar' is allowed and is assumed to depend on a permeability factor $\zeta \in [0,1]$. Thus, the nut oil does not really migrate through sugar, rather ζ is a quantification of the part of oil migration that gradient-driven diffusion cannot account for. The permeability parameter ζ is expected to have an influence on the simulated results, especially since sugar is the most important component of the chocolate in terms of percentage (see Figure 8.3). If the CA-based model is parameterized in the next section using experimental data, the calibrated value for ζ should give an indication of the proportion of oil migration that can be contributed to mechanisms other than gradient-driven diffusion. If these other mechanisms play a minor role, ζ is expected to be close to zero, while if the other mechanisms play a significant role, ζ should be larger.

Table 8.1 gives an overview of the model parameters that need to be calibrated using the experimental data from Section 8.2.

Table 8.1: Overview of model parameters.

Symbol	Description	Range	Unit
P_t	Transition probability	$[0,1]$	-
P_d	Diffusion probability	$[0,1]$	-
ζ	Sugar permeability	$[0,1]$	-
Δt	Physical time for one discrete time step	$]0,29030400]$	s

Finally, a choice has to be made on the manner of applying the transition function, be it synchronously or asynchronously. This is particularly important for the migration part of the transition function Ψ. When opting for a synchronous updating mechanism, the number of conflicts resulting from two cells trying to switch state with the same cell in a certain time step is quite substantial, since there are relatively seen sufficient cells in state nut oil present on the tessellation, especially later on in the evaluation. This entails that not all cells in state nut oil will get the possibility to move during each time step if the transition function Ψ is applied in a synchronous manner. Both a synchronous and an asynchronous updating mechanism can be used for this specific case and an optimized parameter set through fitting to the experimental data can be found. However, the optimized parameter values when using the synchronous updating mechanism will reflect not only the physical meaning associated with the parameters, but also the influence of (many) cells not being able to switch state in a certain time step. The latter hinders the link of the optimized parameters with physical values and process, which is key in this work.

Therefore, both the dissolution and the migration part of the transition function are applied in an asynchronous manner. It is mentioned here that the dissolution part of the transition function could be applied in a synchronous manner, but the choice is made to choose one updating mechanism for the complete transition function in order not to overcomplicate things.

The choice for which type of asynchronous updating mechanisms is driven by computational cost. The use of a certain random order in which cells are updated, whether this order changes after all cells are evaluated or not, requires a list that keeps track of not only all cells that (still) have to be evaluated, but that also keeps track of every change (in position and state) that occurs. Especially the latter incurs a lot of extra costs and is for practical uses of the developed model not desirable. For that reason, the random independent update method is employed here. Although it does not guarantee that every cell is evaluated before a cell is evaluated a second time, the decrease in computational cost and the fact that on average every cell will be evaluated once, justify this option. Figure 8.10 shows a flow chart of the CA-based model for oil migration in chocolate confectionery.

8.4. Results and discussion

8.4.1. Parameterization

The inverse problem of retrieving the values of the four model parameters corresponding to the data presented in Section 8.2, is solved by a grid search of the parameter space. A Sobol sequence is used to generate 2000 points in such a way that a good coverage of the parameter space is obtained [131]. The sum of absolute errors (SAE) for each parameter combination is obtained by comparing the experimental data from Figures 8.5 and 8.6 to the simulated model output at the different sampling times. The simulation results are in fact an average of five repetitions to account for the model's stochastic nature. Here, the SAE is used rather than the more common RMSE as it is less sensitive to outliers, which are clearly present in the

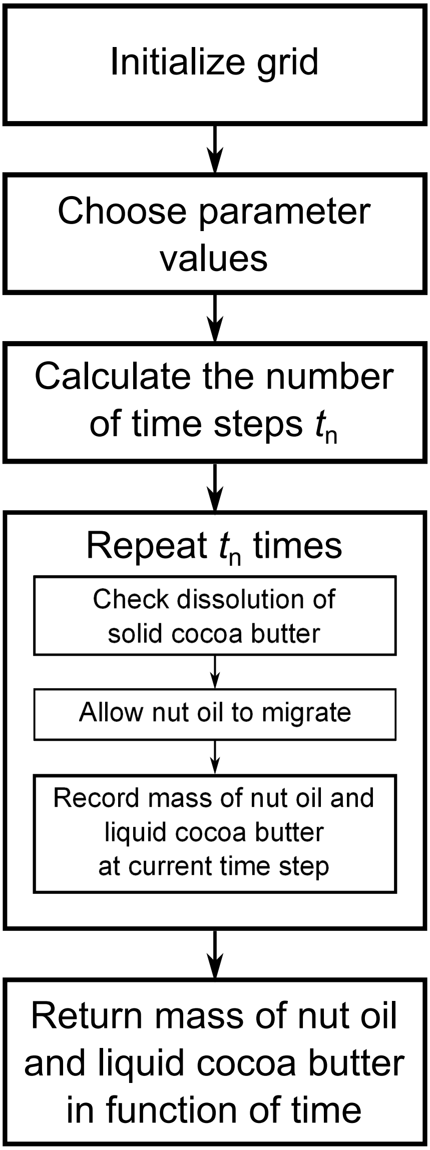


Figure 8.10: Flow chart of CA-based model for oil migration.

experimental data (see Figures 8.5 and 8.6). The main problem that arises from using the RMSE instead of the SAE is the larger relative importance the data points of layer two from week 30 onward play in the parameterization process, causing the simulated results of all other layers to be higher than the experimental results, since the obtained results from all layers are dependent on each other. The tested parameter combination that gives rise to the lowest SAE is listed in Table 8.2. It is pointed out here that the optimal parameter combination from Table 8.2 is only valid at 20 °C, i.e. the same temperature at which the experimental data were gathered. In order to investigate which parameters are influenced by temperature and to what extent, more experimental data at different temperatures are needed.

Table 8.2: Optimized model parameter values.

Parameter	Optimized value	Unit
P_t	0.01776	-
P_d	0.4882	-
ζ	0.8567	-
Δt	348748	s

Figures 8.11 and 8.12 show both the experimental data of layers two, four and six as well as the simulated model output that was obtained using the parameters in Table 8.2. Layer eight is not visualized for the sake of clarity. It can be seen from the figures that there is a good fit for both the nut oil and liquid CB data. From Figure 8.11 the wave of nut oil rising through the chocolate coating can be recognized by the later appearance of nut oil in the chocolate layers further away from the filling. The simulation results of the *in silico* experiments show that for each layer there is a clear lag phase during which no nut oil is present and this period becomes longer as the layer is located further away from the filling. This corresponds with expectations as a gradient is necessary for the diffusive mechanism incorporated in the CA-based model and as this gradient is initially the strongest in the layers closest to the filling.

Further, the simulated total amount of nut oil that migrated into the chocolate coating from the filling in function of time is registered. Figure 8.13 shows the corresponding graph, while Figure 8.14 depicts the same result but as a function of the square root of time. Both figures are in accordance with oil migration profiles reported in literature [125, 127, 132, 133]. It

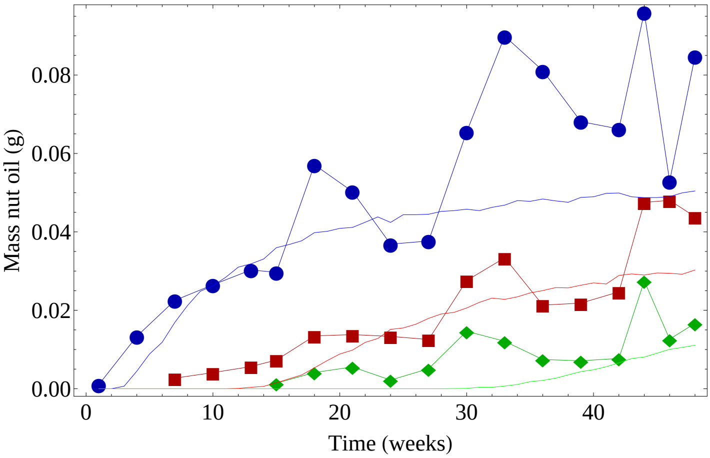


Figure 8.11: Experimental data (with markings) and fitted data (without markings) of the mass of nut oil in function of time for layers two (blue, •), four (red, ▪) and six (green, ◆).

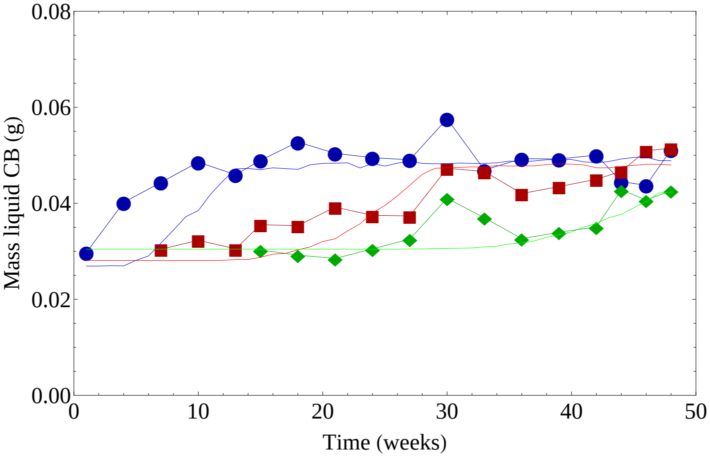


Figure 8.12: Experimental data (with markings) and fitted data (without markings) of the mass of liquid CB fat in function of time for layers two (blue, •), four (red, ▪) and six (green, ◆).

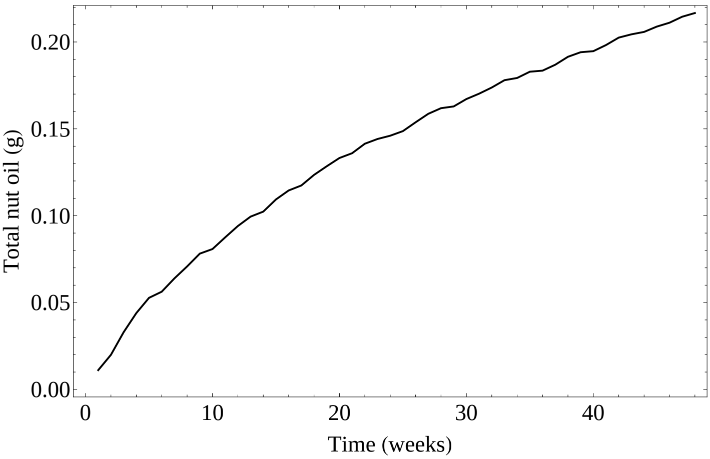


Figure 8.13: Total amount (g) of nut oil in the chocolate layer in function of time.

should be mentioned here that similar profiles to the ones depicted in Figures 8.13 and 8.14 can be found in the field of soil physics to describe the infiltration of water into (heterogeneous) soils [134, 135].

From Table 8.2, it can be seen that the optimal value for Δt is approximately 4 days. This is a realistic value as oil migration is a slow process and therefore changes only become noticeable when studying the system with a large enough time step. This value for Δt, together with the other values from Table 8.2, can also be used to calculate an effective diffusion coefficient, a physical indication of the speed of diffusion. In the considered 2D system, the average distance, i.e. $0.0005 \times P_d$, that nut oil progresses in one time step is $2\sqrt{D\,\Delta t}$. The average distance here is interpreted as the side length of one cell of $\mathcal{T}^*$ multiplied with the probability that this distance is crossed. This results in a diffusion coefficient $D = 4.5 \times 10^{-14}$ m^2/s, which is within the range for D described in literature. The latter range is quite broad and values for the diffusion coefficient range from 2.5×10^{-14} m^2/s [136] to 1.5×10^{-11} m^2/s [137], with our result pertaining to the lower end of the range. For completeness, it should be mentioned that also other parameter combinations gave rise to an SAE close to the optimal one. However, the calculated diffusion coefficient D for all these parameter combinations had the same order of magnitude as the one for the optimal parameter set.

Further, for the parameter combinations giving rise to the lowest SAE, the

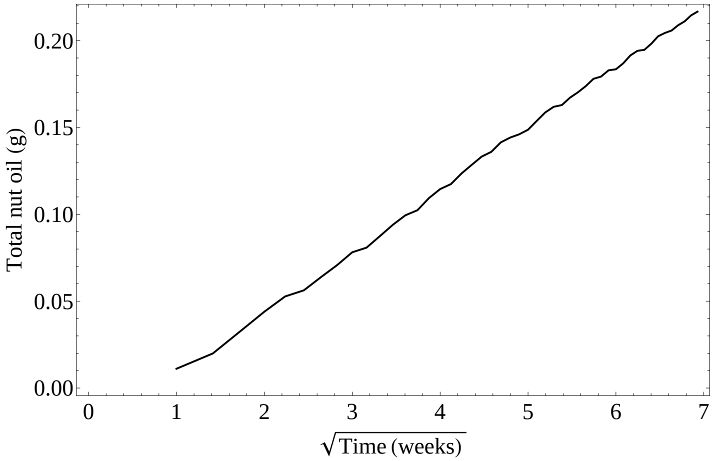


Figure 8.14: Total amount (g) of nut oil in the chocolate layer in function of the square root of time.

sugar permeability ζ clearly has a value different from zero and centered around the optimal value from Table 8.2. This relatively high value indicates that indeed oil migration is more complex than just gradient-driven diffusion and that more mechanisms for oil migration have to be considered. However, the variation of ζ amongst the best parameter combinations is larger than for D, meaning that the simulated model output is probably less sensitive to small variations of this parameter. More and specific experimental data are needed to accurately determine the role of the sugar permeability ζ or, more generally, other mechanisms for oil migration, in the CA-based model. Finally, the optimal value for P_t corresponds to expectations. The low value of P_t indicates that as long as no or very little nut oil is present in the local neighborhood, the dissolution probability P_s is also low and the chemical equilibrium stands. It is only when enough nut oil has migrated into the local neighborhood that the chemical equilibrium is disturbed and that dissolution becomes of importance.

8.4.2. Influence of grid resolution

For practical purposes, modeling results are ideally obtained as fast as possible. One way to achieve this with the CA-based model, would be by respresenting the chocolate confectionery model system by less cells, as such speeding up calculations. However, as mentioned before, this has implications for both the accuracy of the simulated outcome as well as for the parameter sensitivity.

Table 8.3, together with Figures 8.15–8.17, shows the results of the influence of the tessellation resolution on the accuracy of the simulated outcome as well as the speed of the calculations. For each of the studied tessellation resolutions, i.e. 50×260, 40×208, 30×156 and 20×104 cells, the parameterization as explained before is repeated. This means that out of 2000 parameter combinations in the search space, generated by a Sobol sequence, the best parameter combination is sought by looking for the lowest SAE between the experimental and the simulated time series for the migrating fat. The simulation with the optimal parameter combination is repeated 20 times. The SAE that is shown in Table 8.3 is the average SAE over these 20 repetitions and is also represented in Figure 8.15. It can be seen that although the total number of tessellation cells decreases, that the goodness of the fit between experimental and simulated time series stays more or less the same.

Further, as expected, it is clear that having a lower number of tessellation cells substantially decreases calculation times, where the values for the calculation time in Table 8.3 and Figure 8.17 represent the average (out of the 20 repetitions) time needed to complete an evaluation with the CA-based model that corresponds to the 48 weeks during which the experimental data were gathered. On the other hand, it is clear that this increase in calculation speed comes at a price when looking at the standard deviation ξ between the 20 repetitions (see Table 8.3 and Figure 8.16). This standard deviation ξ is an average of the standard deviation at the different measuring times and is almost four times as large for a tessellation of 20×104 cells than for a tessellation of 50×260 cells. The latter implies that although the calculations go faster when the tessellation resolution is 20×104, the results fluctuate much more from simulation to simulation.

Table 8.3: Influence of tessellation resolution on SAE, ξ and calculation time.

Resolution	SAE	ξ	Calculation time (s)
50×260	0.0817	0.00126	213
40×208	0.0752	0.00159	109
30×156	0.0774	0.00247	47
20×104	0.0787	0.00395	17

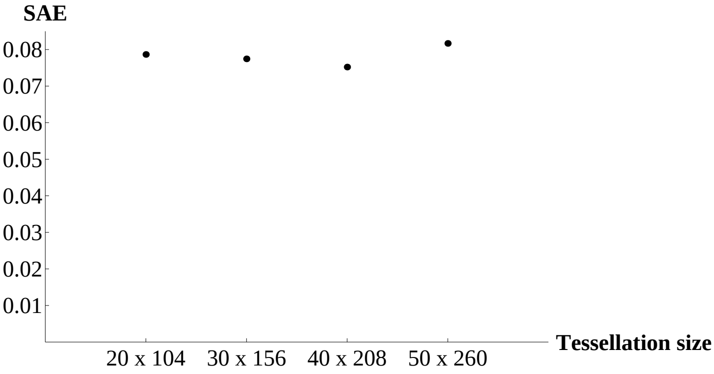


Figure 8.15: Influence of tessellation resolution on SAE.

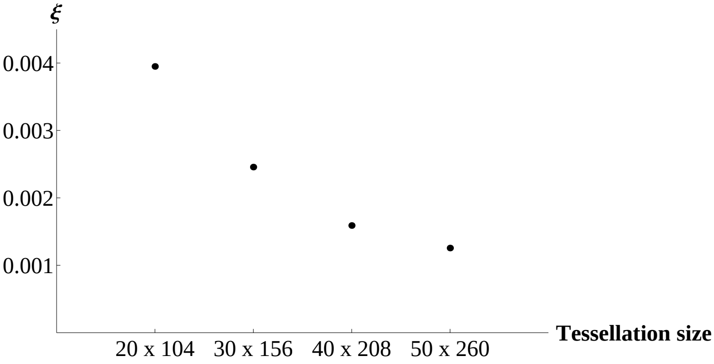


Figure 8.16: Influence of tessellation resolution on standard deviation.

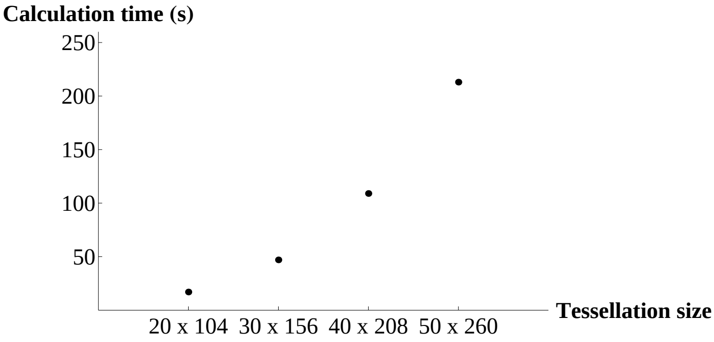


Figure 8.17: Influence of tessellation resolution on calculation time.

Next, the influence of the tessellation resolution on the parameter sensitivity is studied. The EE for each parameter is calculated according to the method described in Section 4.4. Here, a total of 30 trajectories ($\mathcal{R} = 30$) in the search space Ω is chosen. Each trajectory consists of five points: a starting point and a perturbation Δ from this starting point according to each of the b, i.e. four, parameters with $\Delta = 0.02$. The starting points of the $\mathcal{R}$ trajectories are generated according to the Sobol method to obtain a good coverage of Ω. Further, each of the 150 parameter combinations ($\mathcal{R}\,(b+1)$) is repeated 30 times in order to account for stochasticity.

Table 8.4 shows the results of the SA. To facilitate comparison, the table contains the Euclidean distance of a parameter to the origin in the (μ^*, σ) plot as a measure of its sensitivity. Although a clear trend is difficult to discern due to the stochasticity of the model for the three grids with the highest resolution, it is clear that once below a certain threshold resolution, in this case the resolution of 20 × 104 cells, the parameter sensitivity more than doubles. The latter signifies that small deviations from an optimal value for the parameters, results in large differences of the simulated output if the tessellation resolution drops below a certain threshold.

Table 8.4: Influence of tessellation size on parameter sensitivity.

Resolution	H_{P_t}	H_{P_d}	H_{ζ}	$H_{\Delta t}$
50 × 260	1.027	1.363	1.357	1.427
40 × 208	1.223	1.278	1.491	1.361
30 × 156	0.925	1.025	1.243	0.965
20 × 104	2.305	3.120	2.950	2.974

Knowing this, the modeler has to weigh out the advantages and disadvantages of choosing a certain tessellation resolution, depending on the research question.

8.4.3. The importance of capillarity

In order to illustrate the potential of the developed model, the remainder of this section will deal with the incorporation of capillary rise in the CA-based model. Many authors in literature argue that capillary rise plays a role in oil migration in chocolate confectionery (see Section 8.1). However, there is no consensus on how these capillaries appear and how important

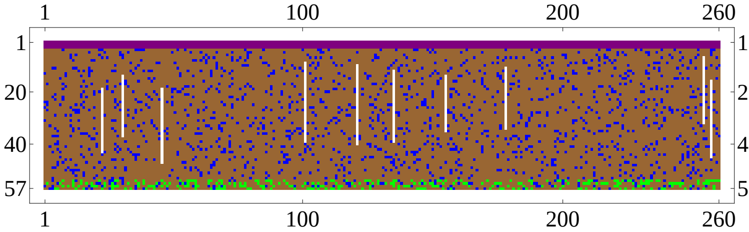


Figure 8.18: Initial condition of the CA, with cavities grouped into capillaries in white, liquid CB in blue, nut oil in green, imaginary cells in purple and the other states in brown.

their presence is. In this setting, model-based hypothesis exploration can provide valuable information. Furthermore, the spatially discrete nature of the proposed CA-based model allows for a straightforward incorporation of spatial information on the chocolate layer if this would be available. Attempts to incorporate spatial information into mathematical models are already being made in the field of soil physics and wood technology [138] and therefore similar approaches could also be applied for studying oil migration in chocolate confectioneries.

To mimic the presence of capillaries, it is assumed that the cavities are grouped to form vertical capillaries through which nut oil can migrate instead of distributing the cavities randomly. These capillaries could be the result of rising air bubbles captured in the chocolate layer during the manufacturing process. Ten capillaries with an average length of 26 cells (i.e. 2% of the total number of cells in the tessellation of size 50×260 cells) were randomly distributed across the cross-section of the chocolate coating [126]. Figure 8.18 shows the initial condition of the CA with the cavities grouped into capillaries. It is noted here that the situation as presented in Figure 8.18 exists only at the beginning of each time step and is therefore only a snapshot of an infinitesimally small moment in time. During each time step, the capillaries are filled with liquid fat migrating upwards.

Capillary rise is described by the Lucas-Washburn equation [139]:

$$\frac{2}{r}\,\alpha\cos\theta = \frac{8}{r^2}\,\omega\,l\,\frac{dl}{dt} + \rho\,g\,l, \tag{8.3}$$

where l (m) is the vertical rise of nut oil in the capillary, α (N/m) the surface tension of the fluid, θ (°) the contact angle between the nut oil and the capillary wall, r (m) the radius of the capillary, ρ (kg/m^3) and

ω (Pa s) the density and viscosity of the fluid, respectively, and g (m/s^2) the acceleration due to gravity. Eqs. (8.4) and (8.5) show the solution of Eq. (8.3) for l.

$$l = l_\infty \left[1 - \exp\left(-\frac{\rho\, g\, r^2}{8\, \omega\, l_\infty} \right) t \right], \tag{8.4}$$

$$l_\infty = \frac{2\, \alpha \cos\theta}{\rho\, g\, r}. \tag{8.5}$$

When making use of the model variables and physical values reported in Altamiras *et al.* [140] and Aguilera *et al.* [125], the maximum capillary rise of nut oil during Δt can be calculated. This rise is far greater than the length of the model capillaries. Therefore, it is assumed that nut oil entering a capillary will reach the top cell of that capillary during the same time step.

As can be seen from Figure 8.11, the lag phase where no nut oil is present is not in agreement with what can be inferred from the experimental data. The data show that there is already a small amount of nut oil present in the upper layers of the chocolate coating early on in the experiment. However, the presence of this nut oil cannot be explained by gradient-driven diffusion as this process is much slower. The refinement of the proposed CA-based model by adding the capillaries could help to explain the traces of nut oil in the upper layers in the early stages of the experiment.

Figure 8.19 shows the state of the CA cells after 40 time steps (± 23 weeks) when the cavities are grouped into capillaries, with initial state given in Figure 8.18 and using the parameter values from Table 8.2 for model evaluation. It can be seen that traces of nut oil (green cells) have already reached the upper layers of the chocolate coating before nut oil cells driven by a diffusion gradient have. Nevertheless, more and specific data are needed to thoroughly investigate the role of capillary rise in oil migration.

8.5. Conclusions

In this chapter, a CA-based model was proposed to describe oil migration in chocolate-coated confectionery. The intuitive nature of such models led

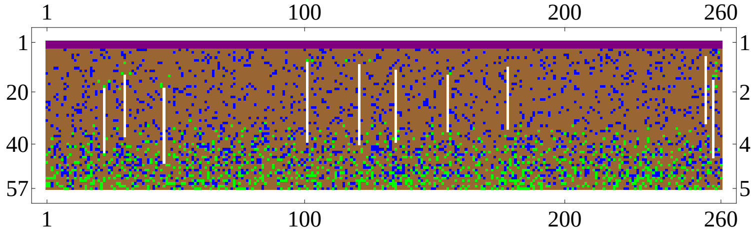


Figure 8.19: Representation of the CA after 40 time steps, with cavities grouped into capillaries in white, liquid CB in blue, nut oil in green, imaginary cells in purple and the other states in brown.

to the construction of a transition function based on the principles of oil migration found in literature. This CA-based model was parameterized using experimental data such that *in silico* results were satisfactory. Further, the model was used to calculate an effective diffusion coefficient as well as to study the importance of oil migration mechanisms different from gradient-driven diffusion through the parameter ζ.

However, more than just an adequate alternative to existing methods, the CA-based approach offers great potential to tackle various kinds of problems. This was demonstrated by performing hypothesis exploration with the introduction of capillaries. Alterations of the transition function or of the structure of the chocolate coating can be readily implemented in the model. Instead of using one general state for all TAGs, one could opt to assign different states to different TAGs if these would differ greatly in physicochemical properties and have a great relative importance within the composition of different TAGs. Each TAG will have its own P_t, P_d and ζ. Therefore, the availability of sufficient and appropriate experimental data is vital in order to introduce different discrete states for different TAGs. The spatial nature of CA-based models also allows for embedding structural information, such as tomographic images. When tomographic images of chocolate confectionery are available, they can be used to more precisely determine the initial state of the CA. In this way the effect of e.g. insufficient mixing of the ingredients in the production stage can be studied. Further, the CA-based model can also keep track of the amount of nut oil that reaches the surface as a function of time. This information, coupled with fat bloom scores by an expert panel and other data and measurements on fat bloom could be of use when studying the moment of onset and the intensity of fat bloom.

It is important to note that although like in the previous cases a 2D tessellation was used, this time, the tessellation represents a cross-section of the complete system under investigation, rather than a small part. The latter enforces the modeler to carefully choose a tessellation resolution, as was shown in Section 8.4. Further, the tessellation representation also has an impact on the boundary conditions. Normally, fixed boundary conditions should be adhered to for all boundaries of the tessellation since the real system is also bounded. However, since no experimental data is available in the horizontal direction but only per vertical layer, the fact that periodic boundary conditions are applied in the horizontal directions does not interfere with the parameterization of the CA-based model. Nevertheless, the periodic boundary conditions do entail an easier model implementation which is the reason they are employed in the horizontal direction for this case. A final remark on the tessellation is that the filling layer is represented by just four rows of which the cells are reset to their initial state after every time step. The latter suffices to simulate a nut oil source that is, for the time range of the experiments considered, infinite, without creating the necessity to model the complete filling layer. The latter would not only be very time consuming, but moreover, would not yield new information since no experimental data of the filling layer are available. Further, the migration of components from the chocolate layer to the filling layer is not taken into account because of the lack of experimental data of the filling layer and also because earlier experiments have shown that this migration is negligible.

Another difference with the previous case study is that albeit more than one time series of data (two in this case) is used to retrieve one single optimized parameter set, in this case the time series do not constitute the same process at different concentrations, but two time series originating from two different processes. Therefore, the optimized parameter set has to correctly give rise to two different types of output simultaneously. One of the parameters that had to be optimized was Δt. This is due to the fact that there is no value available for the diffusion coefficient D under these circumstances. The latter implies that although a choice was made for the tessellation size Δx, D was not able to link it to a value for Δt. After parameterization of the CA-based model, the optimized value of Δt could be used, together with the value of Δx, to calculate an effective diffusion coefficient D for this specific case.

Finally, it is mentioned that the choice to apply the transition function in an asynchronous manner is based on the fact that synchronous updating would result in an excessive number of conflicting movements as such influencing the parameterization to an unacceptable extent for interpretation.

9 Pitting corrosion

Pitting corrosion is difficult to detect, predict and design against. Modeling and simulation can help to increase the knowledge on this phenomenon as well as to make predictions on the initiation and progression of it. A CA-based model describing pitting corrosion is developed in this chapter based on the main mechanisms behind this phenomenon. Further, a sensitivity analysis is performed in order to get a better insight in the model, after which the information gained from this analysis is employed to estimate the model parameters by means of experimental time series for a metal electrode in contact with different chloride concentrations.

9.1. The phenomenon

Corrosion of metals is a natural process, because just like water flows to the lowest level, all natural processes tend toward the lowest possible energy state. To produce metals, starting from naturally occurring minerals and ores, energy needs to be added. It is therefore only natural that when these metals are exposed to their environments, they would revert back to the original state in which they were found [141, 142]. Crucial in the occurrence of corrosion is the combination of the material and the environment, since the corrosion behavior of a material depends both on the environment to which it is subjected and on the metal or alloy composition [141]. In the case of unfavorable combinations, severe corrosion damage to the metal appears.

The effects of corrosion in our daily lifes are both direct, in that corrosion affects the useful service live of our possessions, and indirect, in that producers and suppliers of goods and services incur corrosion costs, which they pass on to customers. The total cost of damage caused by corrosion and the efforts in fighting its destructive effects amounts to 1% - 5% of a country's GNP, of which 10% to 40% can be avoided, mainly by broader application of corrosion-resistant materials and the application of best corrosion-related technical practices [141, 143]. Besides economic issues, there are also safety hazards involved such as contamination by chemical processing plants or

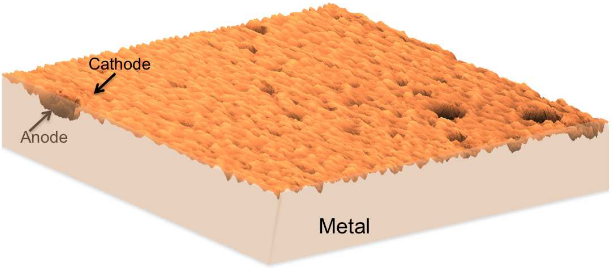


Figure 9.1: Pitting corrosion with indication of anodic and cathodic half-reactions.

construction collapse [142].

One of the forms of corrosion that can be identified based on the morphology of the corroded metal is pitting corrosion [141]. This form of localized corrosion is restricted to small areas and can be recognized by the appearance of small holes on the metal surface as shown in Figure 9.1 [144, 145].

The first step in the pitting corrosion process is the passivity breakdown and the initiation of a localized attack, which can be brought about by various mechanisms [146, 147]. Firstly, the environment of the metal can possess several critical conditions leading to the breakdown of the film in small areas, while preventing dissolution of the entire surface cover [148]. Secondly, the breakdown of the passive film can occur due to special features of the cover itself. Films are generally crystalline or get crystalline with time such that local thinning mechanisms, related to the nanostructure of the barrier layer that consists of nanograins separated by grain boundaries, render these regions susceptible to the pit nucleation process [146]. Besides crystalline grains in the passive film, other point defects are assumed to be electrons, holes, and oxide vacancies [149]. Finally, local depassivation can also be promoted by the presence of impurities or irregularities on the metal surface. All metals or alloys have some physical or chemical inhomogeneities that make them more susceptible to attack in aggressive environments compared to the remaining surface [150]. The chemical composition of the surface will affect the protective properties of passive films more severely than physical defects in the film [150]. These chemical inhomogeneities can be boundaries between the metal matrix and inclusions or between second phase precipitates [151]. Therefore, pits can nucleate at grain boundaries [150], on the grains itself [152], at mechanical damages [153], in heat-affected weld zones [154], dislocations [155] and at defects or inclusions (e.g. MnS, oxides, sulfides, silicates, precipitates of carbides and carbonitrites) in the microstructure of many metals or alloys [156]. When the pitting corrosion is promoted by the presence of inclusions in the metal surface [150, 151], the shape of the pits can be modified by them as shown by Vignal *et al.* [156] and Gahari *et al.* [157]. In the latter, for example, the authors showed that the orientation of an MnS inclusion can be important for the pit evolution.

After the pit initiation in these regions, the acidity inside the pit is maintained by the spatial separation of the cathodic and anodic half-reactions, which creates a potential gradient and electromigration of aggressive anions into the pit [158]. As pit growth progresses, different solution compositions develop inside the cavity and the consequent voltage (IR) drop along the metal/electrolyte interface dictates that the deeper the pit is, the lower the pit growth rate is [159, 160, 161]. In addition, the formation of a lacy metal cover over a growing pit could occur. This cover provides a diffusion barrier which stabilizes the pit growth since it keeps the bottom of the pit in active dissolution [162].

Distributions and characteristics of pitting sites on metal surfaces have been determined through microscopic inspection [163, 164]. The metal surface apparently has a fixed number of these inclusions, i.e. sites for pit nucleation, as observed in recent works by Punckt *et al.* [163] and Zimer *et al.* [164]. The absence of pit creation nearby already existing pits is another issue that is studied. Reuter and Heusler [155] showed that the probability of finding no pit decreases exponentially with the area around an active pit. This produces an exclusion zone of a few μm around each active pit in which new pits cannot be nucleated [155]. González-García *et al.* [165] detected anodic and cathodic current transients on 304 grade austenitic stainless steel by scanning electrochemical microscopy measurements. The microscope tip, set to detect the Fe^{2+} in the anolyte from the metastable pits, detects the reduced background of cathodic current after the anodic transient associated with the pit itself.

A better understanding of the pitting corrosion phenomenon is necessary to combat its destructive effects [141, 143]. Even a small pit with minimal overall metal loss can lead to the failure of an entire engineering system because of other corrosion types that can stem from pitting corrosion such as stress corrosion cracks [142, 150, 166]. Therefore, pitting corrosion has been widely studied for many years. Nevertheless, some aspects of this phenomenon remain unclear. Modeling and simulation enables us to understand and predict the nature and intensity of corrosion, decreases the need for difficult experimental measurements of corrosion under different electrochemical conditions and allows for extrapolating over longer time scales and to other physical conditions. However, the modeling of the corrosion phenomenon is challenging due to its complex nature and the involvement of many variables and consequently up till present, no satisfactory, validated models exist [159].

In the second half of the 20th century, initial corrosion models, mostly based on (P)DEs, were used to describe either the initialization of corrosion [167, 168] or the propagation of corrosion [169, 170, 171]. Later on, models were introduced incorporating both steps, more processes such as passive film formation were added and specific models for specific conditions were developed. Although these models are elaborate, they require extensive knowledge about the material, its origin and the process, which poses problems for both their development and more so for their validation [172]. Therefore, the use of new modeling paradigms such as artificial neural networks [173], statistical models [174] and CAs are being explored [175].

In this chapter, a 3D CA-based model describing pitting corrosion is developed. The use of CAs in the field of corrosion is relatively new [176, 177], but the number of researchers employing CAs as well as the knowledge on the subject is growing. Although many different aspects that influence corrosion, such as film formation, pH, potential differences and heterogeneous composition of alloys, are being explored and are incorporated in CA-based models, most of them only establish a qualitative resemblance between the simulated model output and the real-world phenomenon, ignoring the importance of a sound model validation that is a prerequisite to have a model with predictive value [144, 159, 161, 178, 179, 180, 181].

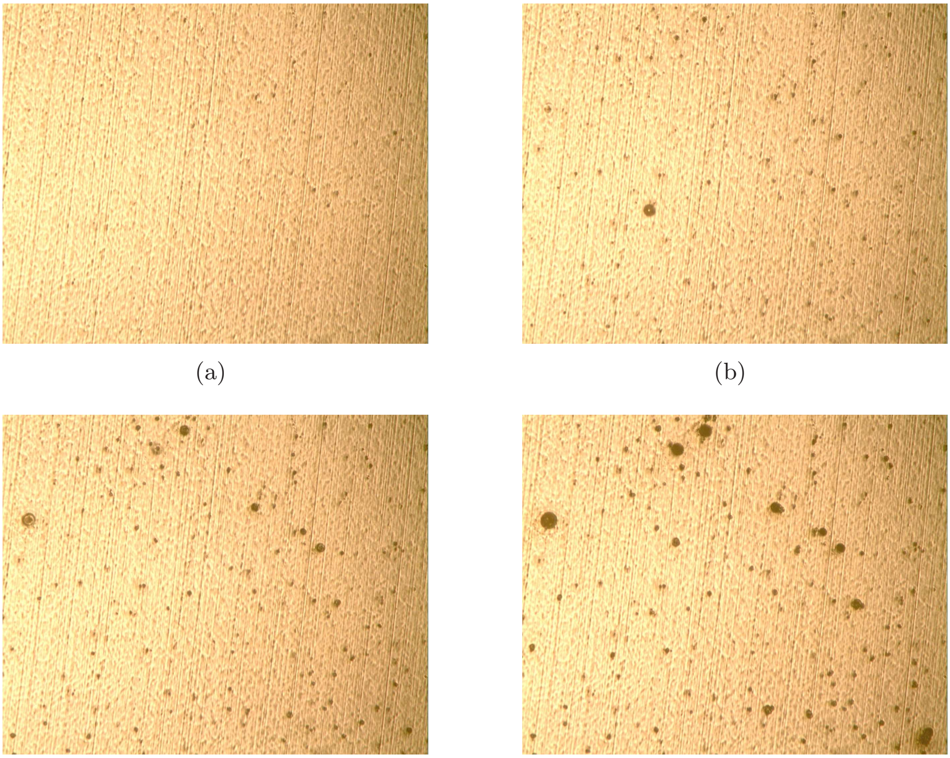
Only very recently, a few authors validated their CA-based model using time series of data, but with modest success [10]. Furthermore, CA-based models in literature are usually two-dimensional, making them unsuited to model the growth of corrosion pits in the direction of the pit depth and surface simultaneously [161]. Three-dimensional models could provide new information about dynamic processes like pit coalescence or the formation of channels and peninsulas inside the metal. Nevertheless, the simultaneous study of all factors involved in pitting corrosion is a complex task to overcome using a single approach. For that reason, the model introduced in this chapter focuses on mass transport, IR drop, pit initiation, metal dissolution and cathodic protection, but leaves aspects such as passivation and bimetallic corrosion with spatially different behavior towards corrosion, aside for the time being.

9.2. Data acquisition

Pitting corrosion on AISI 1040 steel was performed in a hydrogen carbonate solution (0.1 $\mathrm{mol\,dm^{-3}}$) prepared by the dissociation of $NaHCO_3$ (Merck) in deionized water at pH 8.3. Prior to data acquisition, the solution was deaerated for 10 min with N_2. The study of the influence of the chloride concentration during pit initiation and propagation was performed using following weight percentages: 2.25, 2.5 and 3.5 wt. % of NaCl (JT Backer) in the solution. Further, cylindrical steel samples (Sanchelli) with a diameter of 9.5 mm (A = 0.7 $\mathrm{cm^2}$) were used as the working electrode (WE). The WEs were previously abraded with sandpaper up to 2000-grit, polished with diamond paste (1 and 0.25 μm), and degreased in acetone for 1 min in an ultrasonic bath. The material composition is described in Zimer *et al.* [164]. As the reference electrode and auxiliary electrode, Ag/AgCl/KCl (saturated) and a Pt wire were used, respectively.

Two types of electrochemical measurements were performed with an Autolab model PGSTAT 30, being the open-circuit potential (E_{oc}) and chronoamperometric measurements. A homemade flat-bottom cell, previously described in literature [182], was employed in these experiments. The use of this cell enables the coupling of the electrochemical techniques and measurements with *in situ* temporal series of micrographs (TSM) obtained with an inverted optical microscope, brand Opton model TNM-07T-PL. The software tools Scope Photo® 1.0 and MCDE (AMCAP) were used for the data acquisition.

In this approach, the E_{oc} was followed up to its stabilization during 6000 s. After stabilization, all chronoamperometric measurements were performed



(a) (b) (c) (d)

Figure 9.2: Micrographs of the metal surface (680 µm × 544 µm) in contact with a solution with a chloride concentration of 3.5 wt. % at (a) start of experiment, (b) after 15 s, (c) after 50 s and (d) 200 s.

for 1800 s with an overpotential (η) of 350 mV more positive than E_{oc}. At the same time, an area of 680 µm × 544 µm, i.e. 0.52% of the WE, was recorded using an acquisition rate of 0.05 and 1 image per second during the E_{oc} and chronoamperometric measurements, respectively. The optimization of the overpotential η to be applied during the chronoamperometric measurements was investigated at a low chloride concentration of 1.5 wt. %. The value of 350 mV was chosen because it allows for observation of the first pit nucleation when verifying the *in situ* image of the electrode surface. Figures 9.2(a)–9.2(d) show four micrographs of the metal surface at different times during the experiment in contact with a solution with a chloride concentration of 3.5 wt. %.

It is noted here that different areas of 680 µm × 544 µm of the WE electrode were followed in order to assure that this small part of the surface was representative of the whole process occurring during the measurements. Starting at the center of the WE, where the TSM were collected, the electrode surface was studied towards the corners of the sample in the orthogonal directions. Thus, several *in situ* micrographs were collected for these new positions and they were compared with the last frame of the temporal series. For each experimental condition, the spatial distribution of pits on the metal surface in the final frame of the studied time series was similar to the spatial distribution of pits of the rest of the electrode surface, excluding edge effects.

To convert the images (frames) from the TSM into quantitative information, a procedure previously described in literature [164, 183] was employed. The number of pits and total pit area were obtained from the *in situ* TSM. The average pit depth was estimated using a fraction of the total charge consumed during the pit formation. This fraction of total charge changes in every frame according to the pit mouth area observed during chronoamperometric measurements. Finally, a 3D model of pit evolution based on Faraday's law was used to calculate the average pit depth and its evolution over time. Figures 9.3–9.5 show the number of pits, the corroded surface area and the average pit depth, respectively, in function of time and for the different mass percentages of chloride ions in solution. It is noticed from Figure 9.3 that a maximum number of pits is reached for 2.5% chloride in solution instead of for 3.5%. Although not completely elucidated, the explanation seems to lie in a shift in corrosion emphasis dependent on the chloride concentration. At lower concentrations of chloride, more pits are formed than at higher concentrations, but these pits are volume-wise smaller. One possible explanation is that the higher diffusion gradient at higher chloride concentrations forces the chloride cells more towards the inside of the pits. This explains that although 3.5% chloride in solution does not yield the highest number of pits, the total affected surface and the average pit depth at 3.5% chloride is the highest. More details on the procedure and data can be found in Zimer *et al.* [184].

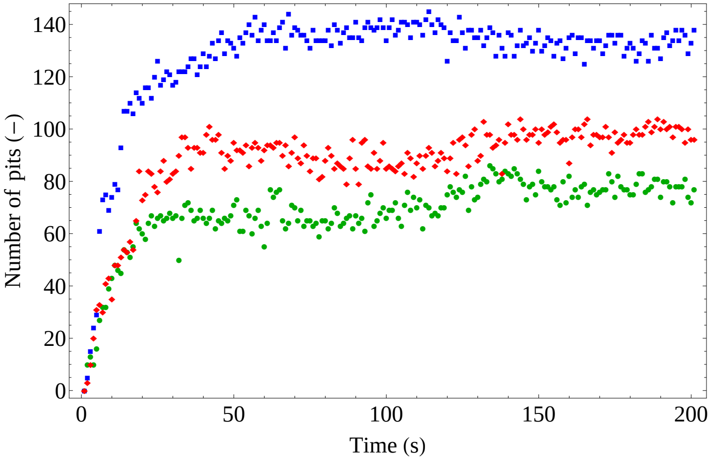


Figure 9.3: Number of pits in function of time for mass percentages of 2.25% (green, ●), 2.5% (blue, ▪) and 3.5% (red, ◆) of chloride ions in corrosive solution.

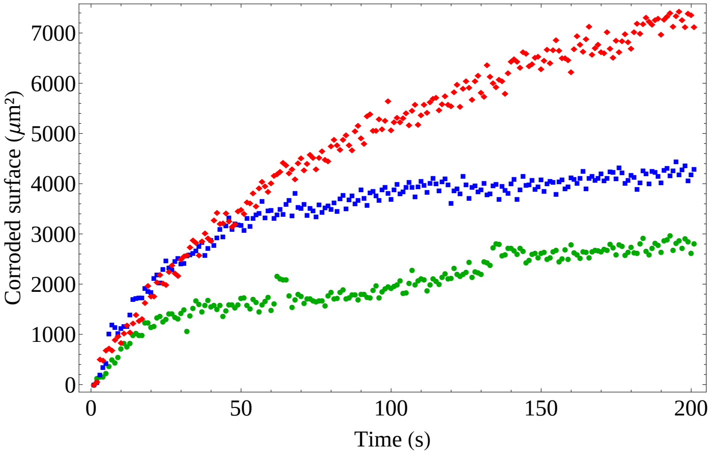


Figure 9.4: Corroded surface area in function of time for mass percentages of 2.25% (green, ●), 2.5% (blue, ▪) and 3.5% (red, ◆) of chloride ions in corrosive solution.

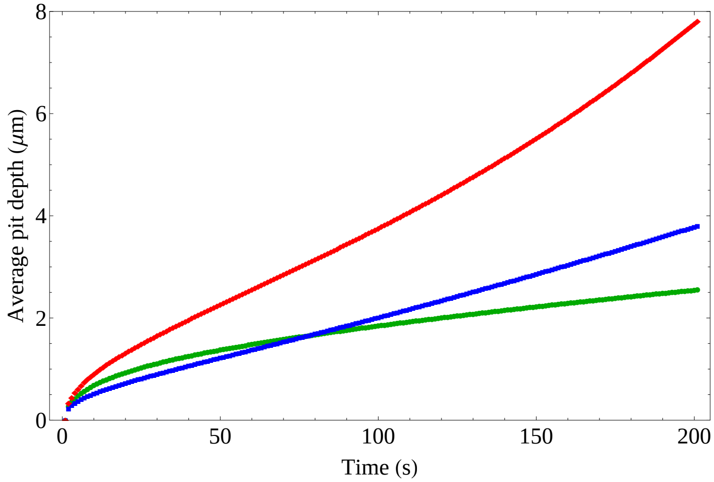


Figure 9.5: Average pit depth in function of time for mass percentages of 2.25% (green, ●), 2.5% (blue, ■) and 3.5% (red, ◆) of chloride ions in corrosive solution.

9.3. Model development

In this section, a stochastic 3D CA-based model describing pitting corrosion is proposed.

A finite 3D tessellation consisting of cubes is used (see Figure 3.6), with i^*, j^* and k^* the number of layers, rows and columns, respectively. Therefore, i represents the thickness direction of the metal, while j and k together form the surface. This tessellation allows for modeling both the affected metal surface as well as the depth of the corrosion pits, both in function of time. Here, a tessellation $\mathcal{T}^*$ consisting of $151 \times 200 \times 200$ cubes is used where the top layer represents the aqueous solution containing the corrosive agent while the other 150 layers represent the metal subjected to corrosion. The number of cells of the tessellation is chosen in this way to obtain a small enough tessellation resolution to prevent overly sensitive model parameters that result in very fluctuating solutions from simulation to simulation, while still not resulting in excessive computation times. Along the j- and k-axes periodic boundary conditions apply, meaning that the cells of the first row (column) are considered to be adjacent to those of the last row (column) to avoid border effects [144]. Along the i-axis on the other hand, fixed boundary conditions are employed since the top layer (aqueous solution) and bottom layer (metal) of $\mathcal{T}^*$ do not physically form an interface and so

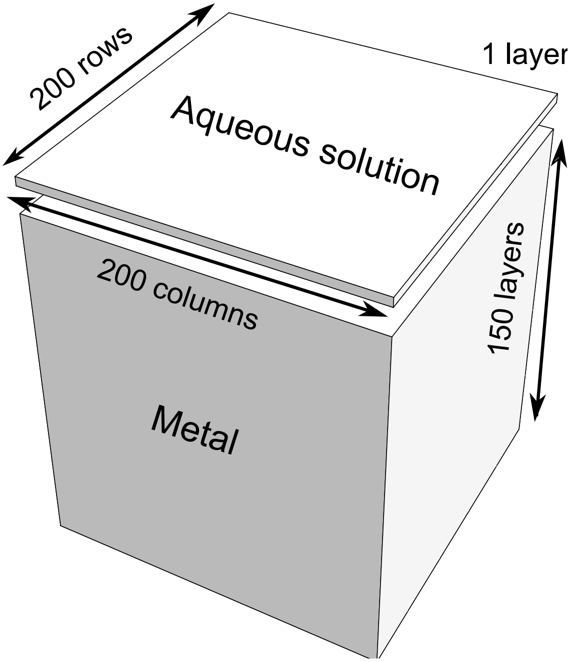


Figure 9.6: Schematic representation of the 3D grid.

through these boundaries no movement is possible.

A schematic representation of the tessellation $\mathcal{T}^*$ is shown in Figure 9.6. The metal part of the tessellation $\mathcal{T}^*$, i.e. the bottom 150 layers, represents only a part of the studied electrode surface used for the data collection to get a resolution fine enough to approximate the experimental results as well as to avoid excessive computation times. By presuming Δj and Δk to be 1.24×10^{-6} m and $\Delta i = 1.44 \times 10^{-7}$ m, the tessellation captures one sixth of the total studied electrode surface area and has a maximum depth large enough to allow for the maximum average experimental pit depth (see Figure 9.5) to occur. The solution in which the metal electrode is submerged is represented here initially by a single layer wherein chloride can diffuse. Since the modeling interest lies in capturing the pitting initiation and propagation processes, in this way the computationally demanding diffusion process is kept to a minimum.

Further, three different discrete states, i.e. $S = \{\text{water, chloride, metal}\}$, are discerned. Although the corrosive solution contains more components than merely water and chloride, here all the non-reactive components are considered as water since they are irrelevant for the current modeling purposes. This chapter deals with a mesoscopic description of the corrosion phenomenon, and therefore cells are not to be associated directly with the individual atoms, anions or cations, but rather to a homogeneous grouping of the same type of atom. The latter means that atomic size effects are not accounted for [185], which is a simple approximation intended to capture the synoptic effects of pitting corrosion [76, 77].

The initial condition of $\mathcal{T}^*$ is determined by the experimental set-up (see Section 9.2) meaning that initially no pits are present in the metal such that all cells of layer two up to layer 151 of $\mathcal{T}^*$ have state metal at $t = 0$. In layer 1, a number of cells is assigned the chloride state according to the mass percentage of chloride in the solution and is randomly distributed across this layer, while the rest of the cells gets assigned the water state.

The transition function Ψ is executed in a stochastic and synchronous manner and falls apart into three parts that describe diffusion, pit initiation and pit propagation, respectively. For the first part of the transition function, only the cells in the chloride state are evaluated, since only their movement is both possible and relevant. Every cell in the chloride state can move once via diffusion in each time step. The motion dynamics of a cell in state chloride are simulated by switching the state of the central cell $c_{i,j,k}$ and the state of the cell in its neighborhood it moves towards. To comply with the law of mass conservation, if more than one cell in state chloride tries to switch state with the same cell, none of them is allowed to switch state, as such employing the adjusted synchronous update mechanism as described in Section 3.3. The complete reaction and diffusion neighborhood N employed in this dissertation is the 3D Moore neighborhood depicted in Figure 3.7(b). However, depending on the position of $c_{i,j,k}$, N can be truncated to fit the fixed boundaries of the system. Initially, a random neighbor of $c_{i,j,k}$ in N is chosen to switch state with after which it is checked whether this cell is in the water state and is therefore a viable candidate to switch state with. Nevertheless, once pits appear on the surface and grow, a concentration and potential gradient emerges and chloride will preferably move (deeper) inside the pits. To capture this, a suction probability $\iota \in [0, 1]$ is introduced that indicates the probability that a cell in state chloride tries to switch state with the lowest possible position in N or if it tries to switch state with any free position in N at random.

Furthermore, since the diffusion coefficient of chloride in these specific conditions is not known, Δt, i.e. the number of seconds corresponding to one discrete time step of the CA-based model, is set as a model parameter. The total number of time steps the CA-based model is denoted t_n and is given by Eq. (9.1):

$$t_n = \lfloor 200/\Delta t \rfloor . \tag{9.1}$$

It is pointed out here that a calibrated value for Δt will not depend on the experimental data alone, but also on the values chosen for Δi, Δj and Δk, since the cell dimensions and the time step cannot be chosen independently and are connected via the diffusion coefficient [179].

The second part of Ψ is the pit initiation where surface metal cells, i.e. where $s(c_{2,j,k}, t) =$ metal, get their state changed to state water or chloride, based on the same ratio as the tessellation initiation, with a probability P_p through an attack of a cell in the chloride state. An attack of a cell in the chloride state starts, like the diffusion part of Ψ, by choosing another cell in its neighborhood N to direct its attack towards. Firstly, it is checked whether the selected cell is a surface metal cell. The next step consists of verifying whether the cell under attack is not already protected through cathodic protection. The latter phenomenon occurs when the metal surface around a newly formed or growing pit becomes negatively charged (see Section 9.1) as such preventing the appearance of new pits in close vicinity of existing pits. The radius of the exclusion zone for each active pit here is taken as 5 μm, based on previous works and observations of the *in situ* images from the metal surface. Translating this cathodic protection to the tessellation $\mathcal{T}^*$, a protection distance of four cells is taken around any existing pit. Finally, an attack only becomes effective with a probability $P_p \in [0, 1]$. From the experimental data (cf. Figure 9.3) a maximal number of pits for each chloride concentration is deduced and when this maximum is reached, P_p is set to 0 from that point on, as such preventing the formation of new pits.

The pit propagation or pit growth, both on the surface as in the depth, forms the third and final part of the transition function Ψ. As for the pit initiation, metal cells get their state changed to water or chloride, based on the same ratio as the initiation of $\mathcal{T}^*$, after an attack by a cell in the chloride state. However, unlike for the pit initiation, the candidate metal cells for attack are metal cells that are part of an already existing pit. In this part of Ψ, all chloride cells choose one of their 3D Moore neighbors at random to attack, and when this neighbor is a metal cell belonging to a pit edge, this metal is dissolved with a probability P_d. There is an indubitable IR control of the current pit growth laws [159, 160] and therefore the IR-factor is incorporated in the CA-based model as a parameter $\upsilon \in [0,1]$. This parameter is used together with the dissolution probability without potential gradient (i.e. the dissolution probability at the metal surface) P_{d0}, again $\in [0,1]$, to calculate P_d in the following manner [159]:

$$P_d = P_{d0} \left(1 - \frac{\upsilon\, w}{w_m}\right), \tag{9.2}$$

with w the depth of the metal cell under attack and w_m the total depth of the metal layer, i.e. 150 cells. From Eq. (9.2), it can be seen that with the IR drop the dissolution probability decreases with increasing pit depth.

Table 9.1 gives an overview of the model parameters that need to be calibrated using the experimental data from Section 9.2 and Figure 9.7 shows a flow chart of the CA-based model.

Table 9.1: Overview of model parameters.

Symbol	Description	Range	Unit
ι	Suction probability	$[0,1]$	-
Δt	Physical time for one discrete time step	$]0,200]$	s
P_p	Pit initiation probability	$[0,1]$	-
P_{d0}	Dissolution probability at metal surface	$[0,1]$	-
υ	IR drop	$[0,1]$	-

9.4. Results and discussion

9.4.1. Sensitivity analysis

The SA results that follow next are obtained using the EE method (for $\mathcal{R} = 30$, $\Delta = 0.02$ and 30 repetitions) and for a mass percentage of chloride ions of 3.5%, but uphold for the two other chloride concentrations. From Figure 9.8 it can be concluded that with regard to the number of pits in function of time, parameters Δt and P_p are the most sensitive, since the

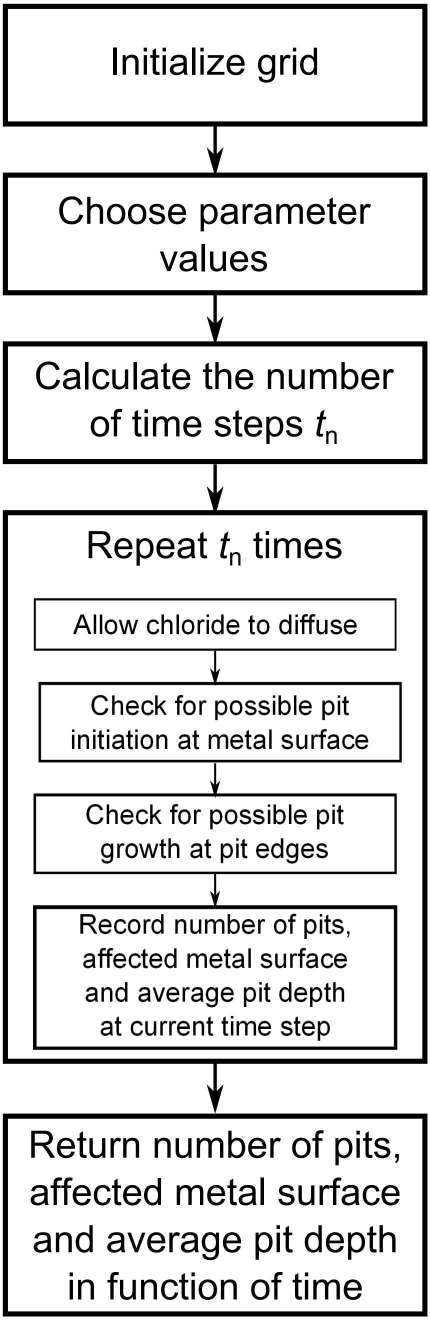


Figure 9.7: Flow chart of CA-based model for pitting corrosion.

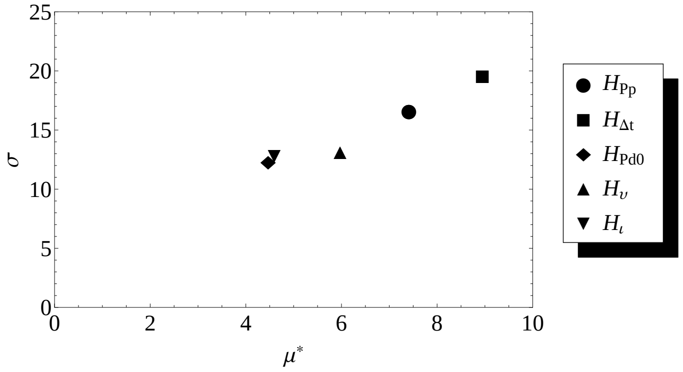


Figure 9.8: Screening of the parameters based on the EE method for the data on the number of pits in function of time for a mass percentage of chloride ions in corrosive solution of 3.5%.

sensitivity measures for these parameters are located furthest away from the origin. This result was expected, because these two parameters together establish the probability of a new pit appearing at a certain moment in time. Analogously, from Figure 9.9 it can be seen that parameters P_{d0} and ι are the ones that most influence the affected metal surface in function of time. This again is a logical outcome since P_{d0} is the probability that a metal cell pertaining to an already formed pit dissolves while ι dictates the preference of a corrosive agent to stay in the neighborhood where it is (by random movements) or move preferentially towards the bottom of the existing pit. Finally, Figure 9.10 shows that the average pit depth in function of time is clearly most sensitive to the value of ι, which could also be anticipated.

9.4.2. Parameterization

The next part of this section deals with solving the inverse problem, i.e. retrieving the values of the model parameters from Table 9.1, corresponding to the observed data from Figures 9.3–9.5 for each of the three different chloride concentrations. This comes down to estimating five parameters, making use of three time series of data (number of pits, affected surface area and average pit depth). Two measures of fitness between the observed data and the corresponding simulated time series with the CA-based model are used. The absolute error (AE) is used between the experimental and

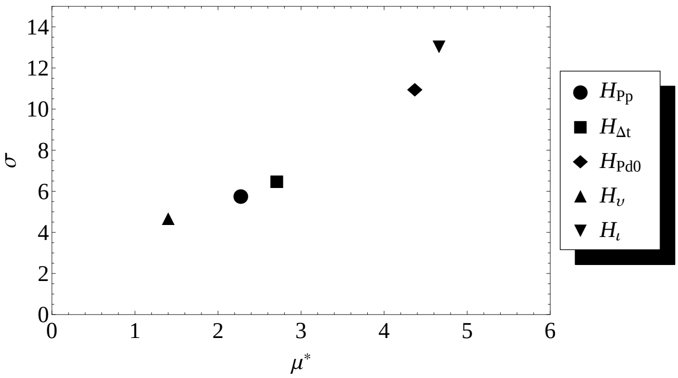


Figure 9.9: Screening of the parameters based on the EE method for the data of the corroded surface in function of time for a mass percentage of chloride ions in corrosive solution of 3.5%.

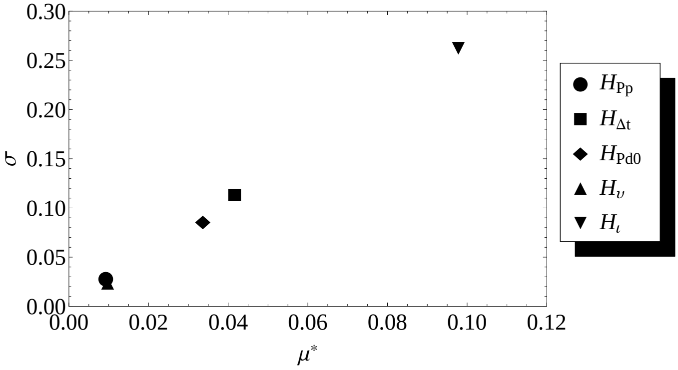


Figure 9.10: Screening of the parameters based on the EE method for the data of the average pit depth in function of time for a mass percentage of chloride ions in corrosive solution of 3.5%.

simulated number of pits in function of time, since this is a discrete measure, and the RMSE is used for the remaining two time series.

Although similar research questions in literature have been tackled by iterative optimization using the Nelder and Mead method and genetic algorithms [10], the optimization method used in this chapter to minimize the AE and the RMSE is a grid search [186], for the same reasons as in the previous two chapters. Initial attempts to estimate all five parameters simultaneously using a grid search have proven unfruitful, probably due to the complex nature of the CA-based model. Therefore, the information gained from the SA is employed during the parameter estimation process to overcome this problem. In a first step, only υ, Δt and P_p are estimated using the time series of the number of pits, while P_{d0} and ι are given a constant value of 0.3. This is justified, since the former three parameters are the most sensitive model parameters for this output and therefore determine predominantly whether the number of pits in function of time can be correctly predicted. Then, in a second step, using the optimal values for υ, Δt and P_p found in the first step, P_{d0} and ι are estimated using the two time series that have not yet been used, being the affected surface area and the average pit depth. In what follows, this approach is elaborated on for a mass percentage of chloride ions of 3.5%.

Some preliminary model exploration allowed to select a subspace of the search space defined in Table 9.1, which is represented in Table 9.2. The subrange of Δt is explained by the fact that values that are too small increase computation times beyond the desirable time a modeler wants to wait for results, while values that are too large are not able to capture the observed data to an acceptable precision. A small value for P_p is also expected since a limited number of pits has to be initiated with a relatively large number of cells in the chloride state being able to attack the metal surface.

Table 9.2: Selected subspace of the search space for the model parameters.

Symbol	Subrange
ι	$[0, 1]$
Δt	$[0.2, 1.2]$
P_p	$[0, 0.001]$
P_{d0}	$[0, 1]$
υ	$[0, 1]$

By means of a Sobol sequence [118], 2000 combinations of υ, Δt and P_p are chosen from their subranges as presented in Table 9.2 in order to get a good coverage of this subspace and as mentioned before $P_{d0} = \iota = 0.3$. An evaluation with the CA-based model is performed with each of the 2000 parameter combinations, the simulated number of pits in function of time is determined, after which the AE is calculated with the experimental data at every point of the time series. Since the proposed CA-based model is stochastic in nature, model evaluations with the 200 parameter combinations that give rise to the lowest AE, are performed in tenfold to account for the stochasticity. Subsequently, from the ten repetitions, the average result is calculated and again the AE is calculated, after which the parameter combination that results in the lowest average AE is selected as the optimal combination. It is worth mentioning that different combinations can result in similar solutions due to equifinality [187]. This is explained by the fact that increasing both Δt and P_p to a certain Δt^* and P_p^* can result in more or less the same simulated output. However, in order to compare the optimal parameter values for different chloride concentrations, a same Δt has to be chosen for all three concentrations. Therefore, a relatively small Δt is selected in order to approximate the observed data for the different concentrations. The obtained optimal parameter values are: $\upsilon = 0.503$, $\Delta t = 0.399$ and $P_p = 0.000201$. Figure 9.11 shows the mean value at each point plus and minus the standard deviation of the ten repetitions and the observed number of pits in function of time.

Next, the optimized parameter values for υ, Δt and P_p are set as a constant for these parameters and 2000 combinations for P_{d0} and ι are chosen using a Sobol sequence. The weighted RMSE is determined for each of the 2000 combinations. This is done by adding up the RMSE that is the result of the comparison between the simulated affected surface area and the experimental data represented in Figure 9.4 and the RMSE that is the result of the comparison between the average pit depth in function of time and the experimental data represented in Figure 9.5. Afterwards, simulations are again performed in tenfold with the 200 parameter combinations that result in the lowest weighted RMSE. It is mentioned here that for reasons of simplicity of the calculations, the data from Figure 9.4 are expressed in the percentage of the metal surface that is affected instead of the surface area in μm^2. The optimal parameter values found here are $P_{d0} = 0.650$ and $\iota = 0.918$ and the simulated outputs using these optimized values are shown

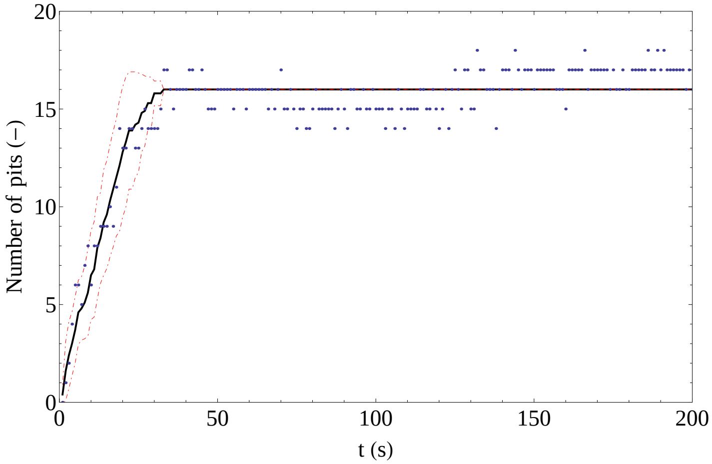


Figure 9.11: Average simulated number of pits (black line) with $\upsilon = 0.503$, $\Delta t = 0.399$, $P_p = 0.000201$, $P_{d0} = 0.3$ and $\iota = 0.3$, the confidence interval around this average output, constructed by the mean value at each point plus and minus the standard deviation of the ten repetitions (red dotdashed lines), and the observed number of pits (blue dots) in function of time for mass percentage of chloride concentration of 3.5%.

in Figures 9.12 and 9.13. It is clear from these figures and Figure 9.11 that although the simulated output corresponds quite good with the observations, further improvement is possible. However, further simulations have shown that when optimizing the parameters for one time series of data, e.g. the number of pits, that a 'good' fit for the other two time series of data is not possible with the model in its current form. Therefore, a parameter combination is chosen that results in a simulated output that gives the best result when all three time series are considered simultaneously.

Figures 9.14(a)–9.14(c) show one of the pits after 200 s as modeled with the CA-based model, i.e. at the end of the experiment. These figures show that the pit grows in a circular shape at the surface, however, this is not the case for all pits that are created during simulations. This is realistic since pits preferentially grow along impurities on the surface, such as small scratches, and are therefore not necessarily circular in shape.

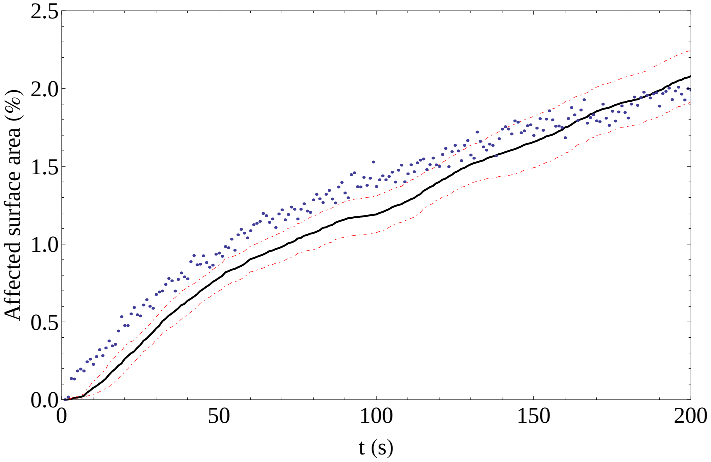


Figure 9.12: Average simulated affected surface area (black line) with $\upsilon = 0.503$, $\Delta t = 0.399$, $P_p = 0.000201$, $P_{d0} = 0.650$ and $\iota = 0.918$, the confidence interval around this average output, constructed by the mean value at each point plus and minus the standard deviation of the ten repetitions (red dotdashed lines), and the observed number of pits (blue dots) in function of time for mass percentage of chloride concentration of 3.5%.

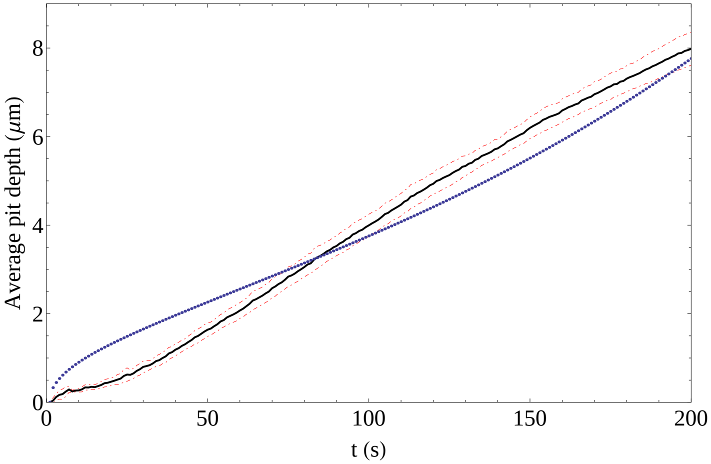
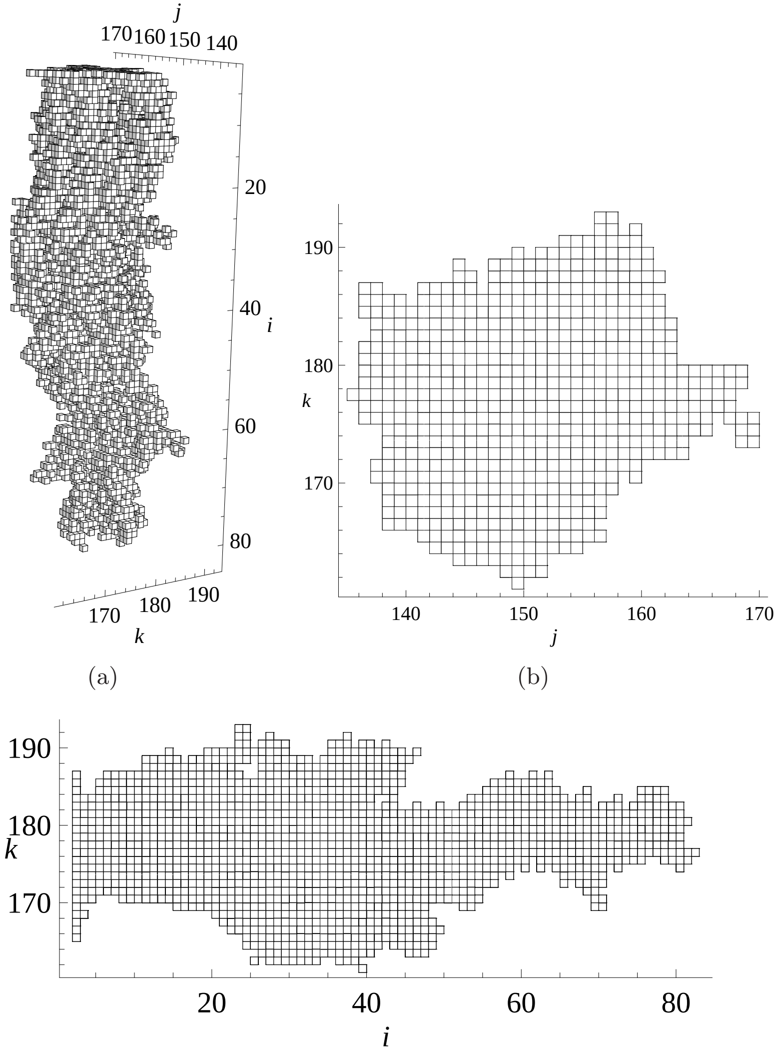


Figure 9.13: Average simulated pit depth (black line) with $\upsilon = 0.503$, $\Delta t = 0.399$, $P_p = 0.000201$, $P_{d0} = 0.650$ and $\iota = 0.918$, the confidence interval around this average output, constructed by the mean value at each point plus and minus the standard deviation of the ten repetitions (red dotdashed lines), and the observed number of pits (blue dots) in function of time for mass percentage of chloride concentration of 3.5%.



(a) (b)

(c)

Figure 9.14: Simulated pit in metal surface after 200 s: (a) Pit in 3D, (b) projection on the $j\,k$-plane (top view) and (c) projection on the $i\,k$-plane (vertical cross-section).

9.4.3. Influence of chloride concentration

The same procedure as before is used to estimate the parameters corresponding to the three time series for the two remaining chloride concentrations, giving rise to the results in Table 9.3. It is clear that the optimized parameter values are dependent on the mass percentage of chloride ions. The suction probability ι is larger for the two highest concentrations since larger concentrations of chloride induce larger concentration and potential gradients, making the influence of this parameter on the diffusion direction of chloride increasingly more important.

The obtained value for P_p increases from a mass percentage of chloride ions of 2.25% to 2.5% and decreases again from 2.5% to 3.5%. The total number of pits in function of time almost doubles going from a mass percentage of chloride ions of 2.25% to 2.5%, which can not be completely attributed to the (small) rise in the mass percentage and therefore a higher value for P_p is necessary. On the other hand, at a mass percentage of 3.5%, both the total number of pits in function of time is lower and the concentration is much higher, resulting in a lower value for P_p again. A similar trend as for P_p can be observed for P_{d0}. The increase in value for P_{d0} from a mass percentage of 2.25% to 2.5% can be explained by the larger affected surface and the higher average depth of the pits as seen in Figures 9.4 and 9.5. Nevertheless, when the chloride concentration rises from 2.5% to 3.5%, the optimal value for P_{d0} drops. The cause for this drop in value is probably due to the combination of the higher chloride concentration on the one hand and the high value for ι on the other hand, keeping the chloride trapped inside the pit and producing the aforementioned results with regard to the affected metal surface and average pit depth.

Finally, although for the three different studied chloride concentrations in the same range, the optimized value for υ shows the same trend as those of P_p and P_{d0}, i.e. rising from 2.25% to 2.5% chloride and decreasing again from 2.5% to 3.5% chloride. This indicates that υ behaves similarly to P_p and P_{d0} to changes in chloride concentration.

9.4.4. Pitting corrosion in bimetals

There are many difficulties associated with the experimental measurements necessary to understand bimetallic corrosion, which is caused by the jux-

Table 9.3: Optimized model parameter values for the experimental data with different chloride concentrations.

	Mass percentage chloride ions		
Parameter	2.25%	2.5%	3.5%
ι	0.703	0.975	0.918
Δt	0.399	0.399	0.399
P_p	0.000278	0.000422	0.000201
P_{d0}	0.150	0.843	0.650
υ	0.586	0.647	0.503

taposition of two or more metals, and as a result a number of theoretical models of the various stages of this and other types of localized corrosion have been developed [148]. Therefore, in order to illustrate the potential of the developed model, the remainder of this section will deal with an exemplary case study for the pitting corrosion of a bimetal.

It is assumed here that the metal block of $\mathcal{T}^*$, i.e. layers two up to 151 in Figure 9.6, is composed out of two types of metal. The first layer of $\mathcal{T}^*$ remains the aqueous layer. Figure 9.15 shows layers two till 151 of $\mathcal{T}^*$ that will be used in the remainder of this section. The central part of the metal block (M1), i.e. columns 51 till 150, is chosen to be a weaker metal, meaning that it is more susceptible to corrosion, than the remaining columns of the metal block (M2). This difference in resistance to corrosion is expressed in the choice of parameter values. Since ι and Δt do not depend on the type of metal, they are chosen to be 0.918 and 0.399, respectively, while for the three remaining parameters a different value is chosen for each type of metal (see Table 9.4).

Table 9.4: Bimetallic parameters for the weaker metal (M1) and the stronger metal (M2).

Parameter	$M1$	$M2$
P_p	0.4	0.2
P_{d0}	0.8	0.65
υ	0.4	0.5

Simulations are performed with the stochastic CA-based model with a mass percentage of chloride ions of 3.5% and with a maximum number of pits equal to 25. Figures 9.16–9.18 show the simulated time series of data for

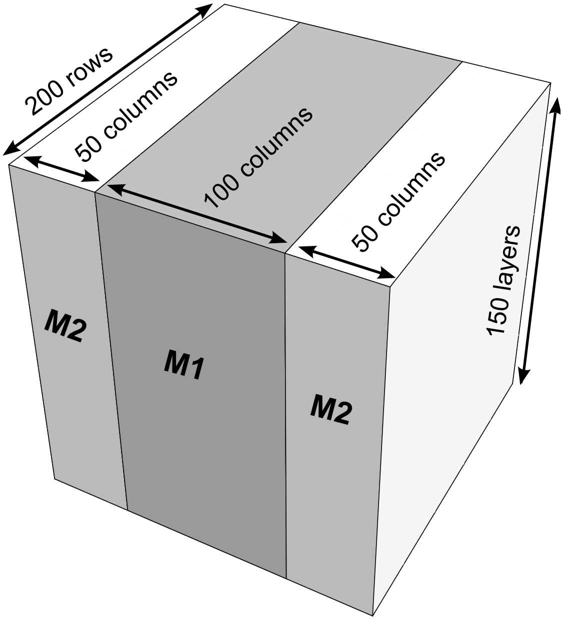


Figure 9.15: Schematic representation of the bimetallic part of the 3D grid.

both types of metal. It can be seen that more pits are initiated on M1 than on M2 and moreover, that the surface of M1 is more affected than that of M2 in function of time. However, it is seen from Figure 9.18 that the average pit depth is larger for pits initiated in M2 than for those that are initiated in M1. This is explained by the fact that with the chosen parameter values, pits in M2 are deeper and more narrow, while pits in M1 are wider and more shallow. Of all the metal cells that are corroded (either by pit initiation or pit propagation), 58% belonged to M1.

Further, when studying the pits that grow in the different types of metal, it can be seen that pits that start in M1 mostly progress in M1 (see Figure 9.19(a)). Pits that initially start in M2 on the other hand, often find a way into M1 and have a branch of the pit in this part as well (see Figure 9.19(b)).

9.5. Conclusions

The developed 3D CA-based model with the incorporation of the key processes of pitting corrosion enables the simulation of pit growth in three

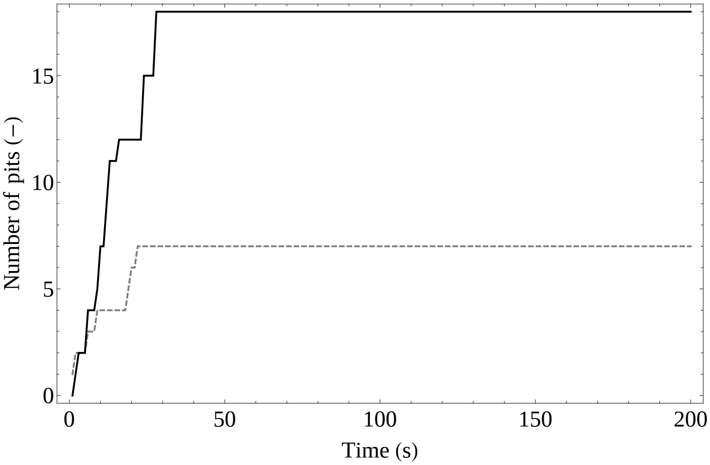


Figure 9.16: Simulated number of pits originating in M1 (black) and M2 (gray, dashed).

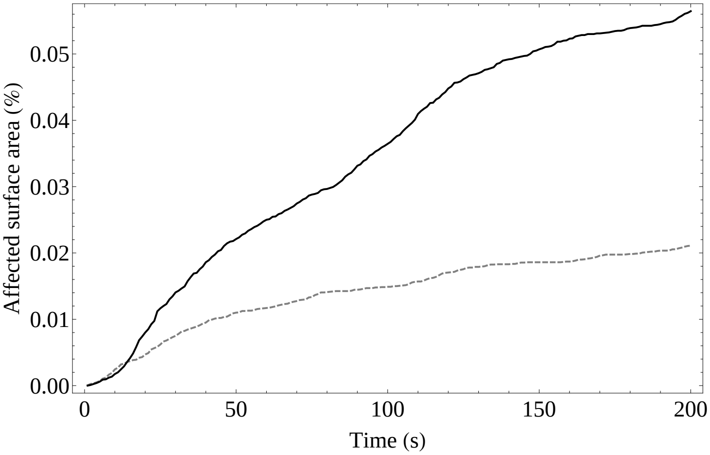


Figure 9.17: Simulated affected surface of M1 (black) and M2 (gray, dashed).

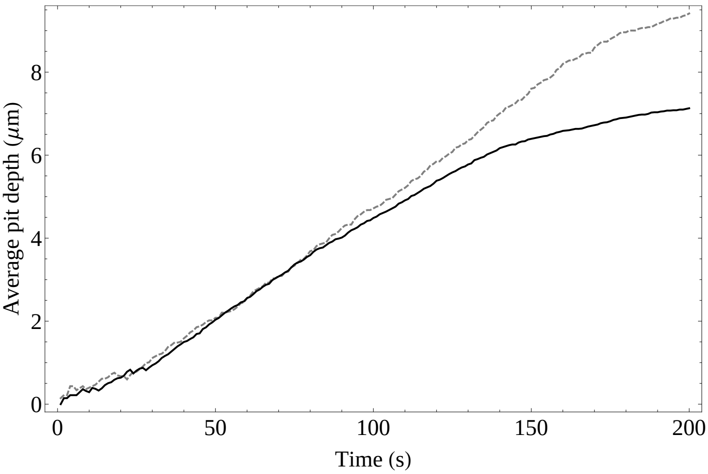
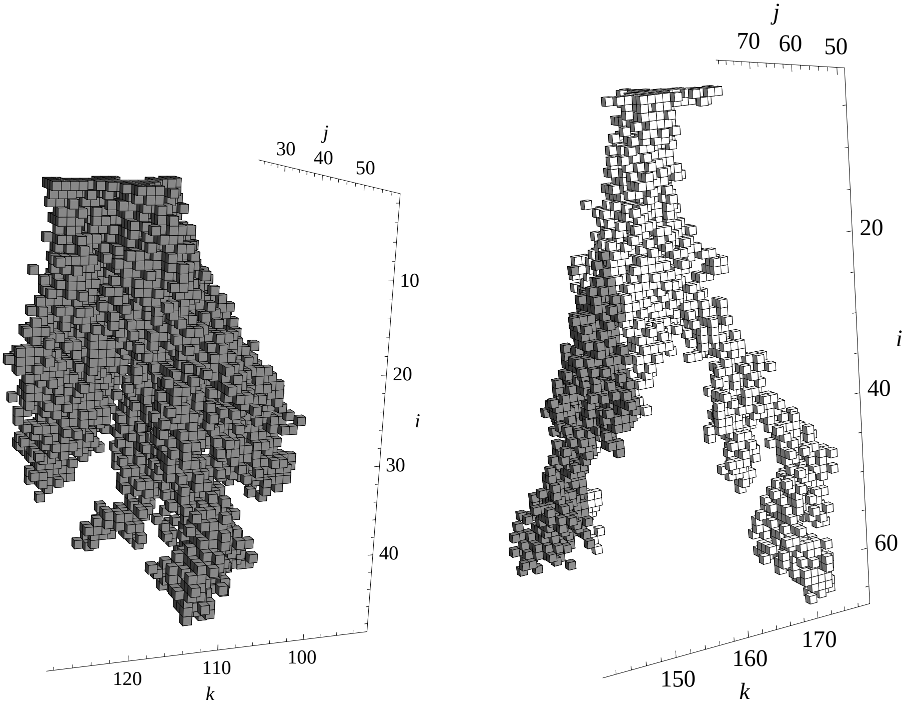


Figure 9.18: Simulated average pit depth of pit originating in M1 (black) and M2 (gray, dashed).



(a)

(b)

Figure 9.19: Bimetalic 3D pit growth of a pit initiated in (a) M1 (gray) and (b) M2 (white).

dimensions. A sensitivity analysis showed that all parameters incorporated into the model are non-redundant and, moreover, the information obtained from this analysis was used to facilitate the parameter estimation. Via a grid search in two steps, the inverse problem of finding a set of parameters that gives rise to three simulated time series as close as possible to the three experimental time series, obtained by the authors in the laboratory, was solved. Further, the influence of the chloride concentration on the calibrated model parameters was investigated. Finally, the potential of the CA-based model was illustrated by means of a case study where pitting corrosion of a bimetalic material was studied.

Nevertheless, it is clear that the fitting of the simulated results to the experimental ones is subject to further improvement. Depending on the degree of accuracy desired of the model, new parts can be added to the transition function of the CA, most likely coupled with the addition of more model parameters, which would possibly improve accuracy, but also make the parameter estimation more difficult. It is therefore crucial to get more insights into the pitting corrosion phenomenon and the CA-based model before adding new parts to the transition function. Model simulations can be used to generate and/or test hypotheses as such increasing the knowledge on the phenomenon. However, experimental data is important to test the validity and necessity of additions and/or changes to the model.

The most limiting point of the model at this moment is the initiation of pits, both in the number of pits as their position and shape. In the current version of the model, the total number of pits is limited by the user, based on the experimental data. Nevertheless, it would be an improvement to the model if the process that stops new pits from appearing, is incorporated into the transition function. If more data were available, this process could possibly be learned from those data. Secondly, the shape of the pit at the surface is more or less circular (see Figure 9.14(b)), although this is not necessarily so in reality. Finally, the positions at which new pits appear are at present randomly distributed over the surface (with the exception of the cathodically protected zone around the existing pits). Therefore, the integration of information on the experimental metal surface would ameliorate the model since new pits are preferentially formed along scratches and other impurities on the metal electrode. Further, it is stressed that the experimental data employed in this chapter stem from a low number of replicated experiments such that little information on the uncertainty of these data is available. Data on more processes and more replications could help to discover new elements to be added to the model as well as to improve the modeling of the processes already incorporated into the model.

For the first time in this dissertation, a 3D tessellation is employed since the experimental data call for it: three dimensions are needed when the surface and the depth of corrosion pits have to be modeled simultaneously. The domain, like was done before, represents only a part of the entire system. However, in contrast with before, periodic boundary conditions are not applied to every boundary. The reason for this is that the tessellation contains an interface of two different components (i.e. aqueous solution and metal) in the vertical direction that cannot be connected through periodic boundary conditions. Therefore, fixed boundary conditions are applied according to the vertical direction, while periodic boundary conditions are used according to the remaining two dimensions, once more simulating the presence of the larger system of which the chosen tessellation is only a part.

Simulating diffusion is a time-consuming task. For this reason, like was done in the case study of oil migration in Chapter 8 with the filling layer, the part of the system where diffusion takes place, but for which no experimental data are available, is kept to a minimum. Here, the choice was made to incorporate just one single layer of aqueous solution in which the corrosive agent diffuses. Obviously, as the simulation progresses and metal is being dissolved, more aqueous solution is introduced in the tessellation. Further, with respect to the diffusion, again Δt is introduced as a parameter since no diffusion coefficient under the circumstances described in Section 9.2 is known.

Finally, the choice for a synchronous application of the transition function is elaborated on. Although the situation in this case study is quite similar to the one presented in Chapter 8, the update of the state of the cells of the tessellation is not performed in an asynchronous manner. The reason for this being twofold. Firstly, the number of cells in state chloride, i.e. the ones that diffuse and attack metal cells, is relatively small in comparison to the total number of water and metal cells. The latter implies that conflicts arising from two chloride cells trying to switch state with the same cell or attempting to attack the same metal cell is limited. Therefore, the influence this choice has on the optimized parameter values is small. The second reason for choosing a synchronous updating mechanism is because of the reduction in computation times it offers, which is of utmost importance when working with 3D models.

PART III

IMPROVED OPTIMIZATION

10 Introduction

Simulations with computer models offer a great time reduction when compared to laboratory experiments. However, the performance of these models strongly depends on the knowledge that is incorporated into the model as well as a correct parameterization of the model [188]. Solving the inverse problem, i.e. the retrieval of optimal values of model parameters from experimental data, remains a bottleneck when modeling (a)biological processes [189, 190]. Also for the different case studies presented in Part II, the parameterization of the developed CA-based models was a complex and time-consuming task. Due to various, mostly unavoidable sources of noise inherent to experimental data, it is impossible to find a set of parameter values for which the model simulations would generate exactly these data, even if the proposed model were the correct one. Therefore, a myriad of optimization algorithms that try to find a near-optimal solution have been developed over the years. Traditionally, the goal of these algorithms consists of finding a set of parameter values that minimizes the (regularized) model error. This error is generally expressed through an objective function.

It is mentioned here that although the interest in the domain of chemical engineering for CA-based models is growing, most of the reported models are neither calibrated, nor validated. This is partly due to the relative newness of this modeling paradigm in this domain and the lack of adequate data to calibrate and validate these models. However, another issue is that the parameterization of CA-based models generally involves non-classical optimization techniques.

Some of the more popular optimization algorithms used for continuous problems are stochastic steepest descent [117], genetic algorithms [191], (simplex) simulated annealing [192, 193] and particle swarm optimization [116]. Although these heuristics have proven to be quite successful in numerous cases, they are not always the best choice as was mentioned already in Part II. Namely, when parameterizing models that give rise to multiple local optima and where a single model evaluation is highly time consuming, unnecessary model evaluations should be avoided and more protection against local optima is desired [114, 115]. Furthermore, when several computer nodes are at disposal for performing model evaluations simultaneously, the (often) iterative heuristics pose a problem, since a calculation is always based on the previous result. In these cases, a grid search [186] that evaluates the model in a number of predefined points in the parameter space, is the preferred procedure, especially when the number of parameters to be determined is not excessively large. The challenge with grid search is efficiently choosing a finite subset of the parameter space for evaluation when a limited number of model evaluations are allowed or desired. The naive approach is to take the same number of parameter values equidistantly according to each of the dimensions of the parameter space, and consider all combinations of these values. However, from the results of the SA of the various case studies it is clear that some parameters are more important to be estimated correctly than others. Therefore, in this part an improved approach for selecting a subset from the parameter space is suggested that makes use of sensitivity measures obtained in earlier stages of model exploration.

In Chapter 11, information is given about the selection of the search grid and how sensitivity information obtained through SA can be employed to improve the selection of a subset of the parameter space for evaluation. Subsequently, in Chapter 12 the proposed approach is exemplified using a simple linear model as well as two well-known nonlinear models and the CA-based model for oil migration in chocolate, introduced in Chapter 8 of Part II.

11 Grid search using sensitivity measures

11.1. Selection of the search grid

11.1.1. Rectangular search grid

It is assumed here that the parameter vector $\boldsymbol{u}$ to be optimized is a vector of two parameters u_1 and u_2 and that the search grid $\mathcal{G} \subset \mathbb{R}^2$ contains $\mathcal{N}$ grid points:

$$\mathcal{G} = \left\{\boldsymbol{u}^j\right\}_{j=1}^{\mathcal{N}} = \left\{\left(u_1^j, u_2^j\right)\right\}_{j=1}^{\mathcal{N}}. \tag{11.1}$$

Let $\boldsymbol{q}^* = (q_1^*, q_2^*) \in \mathbb{R}^2$ be the optimal parameter vector according to some optimality criterion. The Euclidean distance between $\boldsymbol{q}^*$ and a grid point $\boldsymbol{u} \in \mathcal{G}$ can be used to measure the approximation error of $\boldsymbol{u}$, i.e. $\boldsymbol{u}$ has fitness $(u_1 - q_1^*)^2 + (u_2 - q_2^*)^2$. Moreover, the approximation error $d_{\mathcal{G}}$ of a grid $\mathcal{G}$ can be defined as the minimum of the approximation errors of its elements:

$$d_{\mathcal{G}} = \min_{j=1}^{\mathcal{N}} \left(\left(u_1^j - q_1^*\right)^2 + \left(u_2^j - q_2^*\right)^2\right). \tag{11.2}$$

Now, assume that the optimal parameter vector is a random vector $\boldsymbol{u}^*$ that is uniformly distributed over the parameter space. Naturally, the approximation error of $\mathcal{G}$ becomes a random variable as well:

$$D_{\mathcal{G}} = \min_{j=1}^{\mathcal{N}} \left(\left(u_1^j - u_1^*\right)^2 + \left(u_2^j - u_2^*\right)^2\right). \tag{11.3}$$

Thus, $D_{\mathcal{G}}$ represents the smallest Euclidean distance to $\boldsymbol{u}^*$ among the $\mathcal{N}$ search grid points. Further, we say that a grid $\mathcal{G}_1$ performs better than another grid $\mathcal{G}_2$, which is denoted as $\mathcal{G}_1 \trianglelefteq \mathcal{G}_2$, when $D_{\mathcal{G}_2}$ stochastically

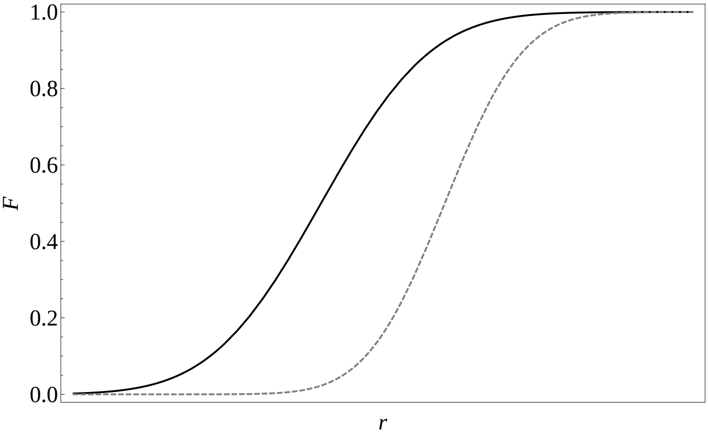


Figure 11.1: Cumulative distribution functions $\mathcal{F}_{D_{\mathcal{G}_1}}$ (black) and $\mathcal{F}_{D_{\mathcal{G}_2}}$ (gray, dashed).

dominates $D_{\mathcal{G}_1}$ [194]. Explicitly, $\mathcal{G}_1 \trianglelefteq \mathcal{G}_2$ expresses that

$$(\forall r \in \mathbb{R}^+)(\mathcal{F}_{D_{\mathcal{G}_1}}(r) \geq \mathcal{F}_{D_{\mathcal{G}_2}}(r)). \tag{11.4}$$

Equation (11.4) is a very strict condition: $\mathcal{G}_1 \trianglelefteq \mathcal{G}_2$ is equivalent to saying that the cumulative distribution function $\mathcal{F}_{D_{\mathcal{G}_1}}$ of the random variable $D_{\mathcal{G}_1}$ has to be greater than or equal to the cumulative distribution function $\mathcal{F}_{D_{\mathcal{G}_2}}$ of $D_{\mathcal{G}_2}$ (see Figure 11.1).

Interestingly, for a given grid $\mathcal{G}$ and using the Euclidean distance to measure the approximation error, evaluating $\mathcal{F}_{D_{\mathcal{G}}}(r)$ boils down to determining the ratio of the area enclosed by circles with radius r around each grid point to the total area of the search space:

$$\mathcal{F}_{D_{\mathcal{G}}}(r) = \mathcal{P}\left(\min_{j=1}^{\mathcal{N}} \left(\left(u_1^j - u_1^*\right)^2 + \left(u_2^j - u_2^*\right)^2\right) \leq r\right). \tag{11.5}$$

When considering the most basic search grid where the grid points form squares that have a length m, hereafter referred to as a square search grid (see Figure 11.2), this ratio can be determined from an elementary part of the grid. Two examples are shown in Figures 11.3(a) and 11.3(b). These figures show that as long as $r \leq m/2$, there is no overlap of the circles with radius r around the different grid points. By calculating the ratio of the

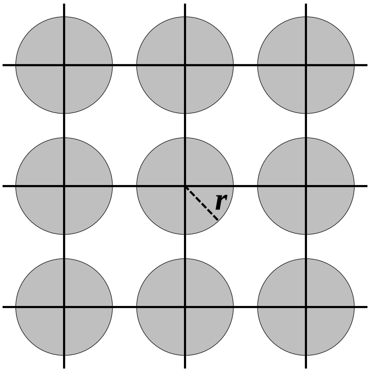
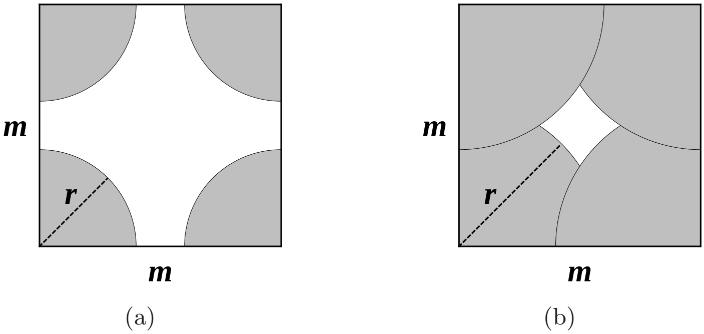


Figure 11.2: Square search grid with indication of the gray area enclosed by circles with radius r around the search grid points.



(a)

(b)

Figure 11.3: Representation of an elementary part of the square search grid with indication of the circle segments with radius r around the grid points for: (a) $r \le m/2$ and (b) $m/2 \le r \le m/\sqrt{2}$.

area of the gray circles to the area m^2 of the elementary part, $\mathcal{F}_{D_\mathcal{G}}(r)$ can be computed as: $\mathcal{F}_{D_\mathcal{G}}(r) =$

$$\begin{cases} \dfrac{\pi\, r^2}{m^2} & \text{, if } r \le \frac{m}{2}, \qquad (11.6a) \\ \dfrac{\pi\, r^2 - 2\, r^2 \left(2 \arccos\left(\frac{m}{2\,r}\right) - \sin\left(2 \arccos\left(\frac{m}{2\,r}\right)\right)\right)}{m^2} & \text{, if } \frac{m}{2} \le r \le \frac{m}{\sqrt{2}}, \qquad (11.6b) \\ 1 & \text{, if } r \ge \frac{m}{\sqrt{2}}\ . \qquad (11.6c) \end{cases}$$

When comparing the square search grid with any other grid, it has to be taken into account that the same number of grid points has to be considered. So, when the square grid is compared with a rectangular search grid, the area of a rectangle has to be equal to m^2. Figure 11.4 shows a possible rectangular search grid where the elementary parts have an area of m^2 and where the gray circles represent the area enclosed by circles with radius r around each grid point. Again, the ratio of the area of the gray circle segments within an elementary part to m^2 can be calculated (see Figures 11.5(a) and 11.5(b)). However, compared with the previous setting, there is now a fourth situation that occurs when the circle segments are already overlapping along the shortest sides of the rectangle but not yet along the longest sides. In case of the rectangular search grid with dimensions as shown in Figures 11.5(a) and 11.5(b), $\mathcal{F}_{D_\mathcal{G}}(r)$ is given by: $\mathcal{F}_{D_\mathcal{G}}(r) =$

$$\begin{cases} \dfrac{\pi\, r^2}{m^2} & \text{, if } r \leq \frac{m}{3}, \qquad (11.7\text{a}) \\ \dfrac{\pi\, r^2 - r^2\left(2\arccos\left(\frac{m}{3\,r}\right) - \sin\left(2\arccos\left(\frac{m}{3\,r}\right)\right)\right)}{m^2} & \text{, if } \frac{m}{3} \leq r \leq \frac{3\,m}{4}, \qquad (11.7\text{b}) \\ \dfrac{\pi\, r^2 - r^2\left(2\arccos\left(\frac{m}{3\,r}\right) + 2\arccos\left(\frac{3\,m}{4\,r}\right)\right)}{m^2} + & \\ \quad \dfrac{r^2\left(\sin\left(2\arccos\left(\frac{m}{3\,r}\right)\right) + \sin\left(2\arccos\left(\frac{3\,m}{4\,r}\right)\right)\right)}{m^2} & \text{, if } \frac{3\,m}{4} \leq r \leq \frac{\sqrt{97}\,m}{12}, \qquad (11.7\text{c}) \\ 1 & \text{, if } r \geq \frac{\sqrt{97}\,m}{12}\,. \qquad (11.7\text{d}) \end{cases}$$

When comparing Eqs. (11.6a)–(11.6c) with Eqs. (11.7a)–(11.7d), it is clear that $\mathcal{G}_{\text{square}} \trianglelefteq \mathcal{G}_{\text{rectangle}}$. Next, Eqs. (11.7a)–(11.7d) are written in a more general form where the length of the long rectangular side is denoted $z\,m$ and the short side m/z, with $z \geq 1$: $\mathcal{F}_{D_\mathcal{G}}(r) =$

$$\begin{cases} \dfrac{\pi\, r^2}{m^2} & \text{, if } r \leq \frac{m}{2\,z}, \qquad (11.8\text{a}) \\ \dfrac{\pi\, r^2 - r^2\left(2\arccos\left(\frac{m}{2\,z\,r}\right) - \sin\left(2\arccos\left(\frac{m}{2\,z\,r}\right)\right)\right)}{m^2} & \text{, if } \frac{m}{2\,z} \leq r \leq \frac{z\,m}{2}, \qquad (11.8\text{b}) \\ \dfrac{\pi\, r^2 - r^2\left(2\arccos\left(\frac{m}{2\,z\,r}\right) + 2\arccos\left(\frac{z\,m}{2\,r}\right)\right)}{m^2} + & \\ \quad \dfrac{r^2\left(\sin\left(2\arccos\left(\frac{m}{2\,z\,r}\right)\right) - \sin\left(2\arccos\left(\frac{z\,m}{2\,r}\right)\right)\right)}{m^2} & \text{, if } \frac{z\,m}{2} \leq r \leq \frac{\sqrt{1+z^4}\,m}{2\,z}, \qquad (11.8\text{c}) \\ 1 & \text{, if } r \geq \frac{\sqrt{1+z^4}\,m}{2\,z}\,. \qquad (11.8\text{d}) \end{cases}$$

Figure 11.6 shows $\mathcal{F}_{D_\mathcal{G}}$ for different values of z and with $m = 1$. It is clear from this figure that $\mathcal{F}_{D_\mathcal{G}}(r)$ for threshold values r between 0 and 1 and for $z = 1$ is greater than or equal to $\mathcal{F}_{D_\mathcal{G}}(r)$ for any other value of z. For that reason, it can be concluded that the optimal value for z is 1 and

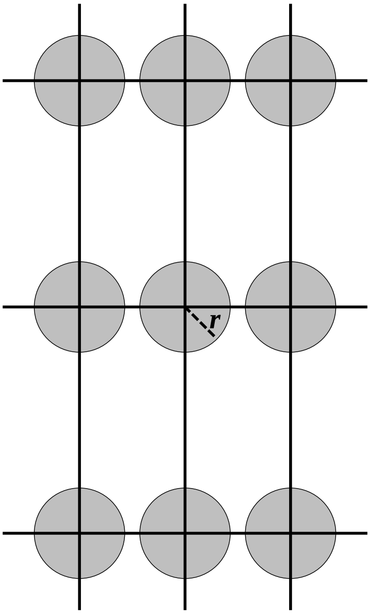
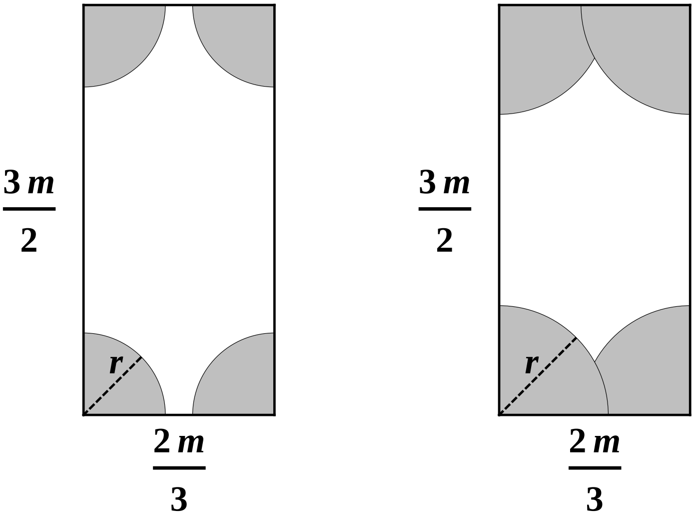


Figure 11.4: Rectangular search grid with indication of the gray area enclosed by circles with radius r around the search grid points.



(a) (b)

Figure 11.5: Representation of an elementary part of the rectangular search grid with indication of the circle segments with radius r around the grid points for: (a) $r \leq m/3$ and (b) $m/3 \leq r \leq 3\,m/4$.

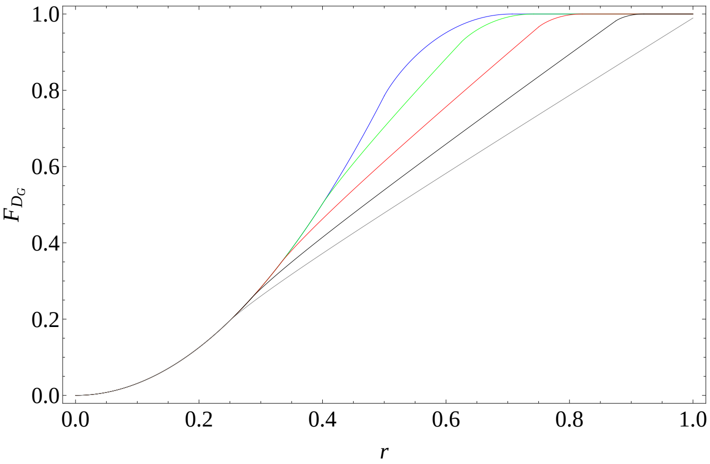


Figure 11.6: $\mathcal{F}_{D_\mathcal{G}}$ with $m = 1$ using Eqs. (11.8b)–(11.8d) for: (a) $z = 1$ (blue), (b) $z = 1.25$ (green), (c) $z = 1.5$ (red), (d) $z = 1.75$ (black) and (e) $z = 2.0$ (gray).

that the optimal ratio of the lengths of the rectangle sides is also 1. The latter implies that when the studied area around the grid points is made up out of circles, the optimal rectangular search grid is the square one. This conclusion can also be directly derived from Eqs. (11.8b)–(11.8d) and is illustrated in Figure 11.6.

So far the Euclidean distance between a candidate set of parameter values and $\boldsymbol{u}^*$ was used as a measure of the quality of approximation. Nevertheless, in practice $\boldsymbol{u}^*$ is unknown and only the observations that are used for parameterization are available, such that another measure of fitness is necessary. A common choice is the sum of squared errors (SSE):

$$R(\boldsymbol{u}) = \sum_i (y_i - f(\boldsymbol{x_i}, \boldsymbol{u}))^2, \tag{11.9}$$

with $\boldsymbol{u}$ the parameter vector, y_i the experimental data, $\boldsymbol{x_i}$ the input vectors and $f(\boldsymbol{x_i}, \boldsymbol{u})$ the simulated model outputs obtained with parameter vector $\boldsymbol{u}$. It is assumed here that there exists only one global optimum $\boldsymbol{u}^*$ where $R(\boldsymbol{u}^*)$ is minimal. Further, for reasons of simplicity, $\boldsymbol{u}$ is still taken as a vector of two parameters u_1 and u_2. Figure 11.7 shows a possible 2D contour plot of $R(\boldsymbol{u})$ for a linear model with indication of $\boldsymbol{u}^*$ as a white dot and with an equal plot range for both parameters.

Although the exact location of $\boldsymbol{u}^*$ is not known beforehand, information

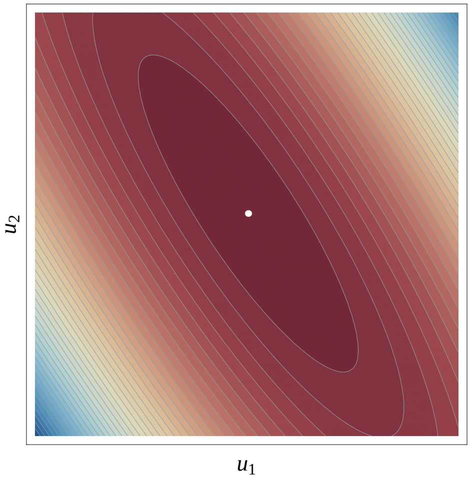


Figure 11.7: 2D contour plot of $R(\boldsymbol{u})$ for a linear model with indication of $\boldsymbol{u}^*$ as a white dot and with an identical plot range for both parameters.

on the contour lines is available via the Hessian H of $R(\boldsymbol{u})$. The first-order partial derivative of $R(\boldsymbol{u})$ to u_1 is calculated in Eq. (11.10) and the second-order partial derivatives of $R(\boldsymbol{u})$ using Eq. (11.10) are presented in Eqs. (11.11) and (11.12):

$$\begin{aligned}\frac{\partial R(\boldsymbol{u})}{\partial u_1} &= \sum_i -2\ (y_i - f(\boldsymbol{x_i},\boldsymbol{u}))\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1} \\ &= \sum_i -2 \left(y_i\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1} - f(\boldsymbol{x_i},\boldsymbol{u})\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1} \right) \qquad (11.10)\end{aligned}$$

$$\begin{aligned}\frac{\partial^2 R(\boldsymbol{u})}{\partial u_1^2} &= \sum_i -2 \left(y_i\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1^2} - \left(\frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1}\right)^2 - f(\boldsymbol{x_i},\boldsymbol{u})\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1^2} \right) \\ &= \sum_i 2 \left(\frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1}\right)^2 + 2\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1^2}\ (f(\boldsymbol{x_i},\boldsymbol{u}) - y_i) \qquad (11.11)\end{aligned}$$

$$\begin{aligned}\frac{\partial R(\boldsymbol{u})}{\partial u_1\, \partial u_2} &= \sum_i -2 \left(y_i\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1\, \partial u_2} - \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1}\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_2} - f(\boldsymbol{x_i},\boldsymbol{u})\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1\, \partial u_2} \right) \\ &= \sum_i 2\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1}\ \frac{\partial f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_2} + 2\ \frac{\partial^2 f(\boldsymbol{x_i},\boldsymbol{u})}{\partial u_1\, \partial u_2}\ (f(\boldsymbol{x_i},\boldsymbol{u}) - y_i) \qquad (11.12)\end{aligned}$$

Analogous derivatives can be computed starting with the first-order partial derivative of $R(\boldsymbol{u})$ to u_2, such that the Hessian is given by:

$$H(R) = \begin{pmatrix} \dfrac{\partial^2 R(\boldsymbol{u})}{\partial u_1^2} & \dfrac{\partial R(\boldsymbol{u})}{\partial u_1 \partial u_2} \\ \dfrac{\partial R(\boldsymbol{u})}{\partial u_2 \partial u_1} & \dfrac{\partial^2 R(\boldsymbol{u})}{\partial u_2^2} \end{pmatrix}.$$

For a linear model

$$f(x, \boldsymbol{u}) = u_1 + u_2\, x, \tag{11.13}$$

$H(R)$ can be simplified in the following manner:

$$H(R) = \begin{pmatrix} 2\sum_i x_i^2 & 2\sum_i x_i \\ 2\sum_i x_i & 2\sum_i \mathcal{N} \end{pmatrix}.$$

It is clear that for the special case of the linear model in Eq. (11.13) where $2\sum_i x_i = 0$ and $\sum_i x_i^2$ equals the total number of grid points, the contour lines are indeed circles as discussed previously. When $\sum_i x_i^2 \neq \mathcal{N}$, the contour lines are ellipses that have axes parallel to the Cartesian coordinate axes. The same calculations as for the circular contours on a regular rectangular grid can be repeated with ellipses. Here, a ratio of the lengths of the ellipse axis parallel to the vertical grid axis to the one parallel to the horizontal grid axis equal to two is selected. Figure 11.8 shows the area enclosed around the grid points for these ellipsoidal contours for a square search grid. The ratio of the area covered by the ellipsoidal contours to the total grid area in function of the threshold r and where the length of the long rectangular side is again $z\,m$ and the short side m/z ($z \geq 1$), can again be determined from an elementary part of the grid (see Eqs. (11.14a)–Eqs. (11.14e)).

$$
\mathcal{F}_{D_{\mathcal{G}}}(r) = \begin{cases}
\dfrac{\pi\, r^2}{2\, m^2} & ,\ \text{if}\ \ r \leq \frac{z\, m}{2}\ \text{and}\ r \leq \frac{m}{z}, \qquad (11.14\text{a}) \\
\dfrac{\pi\, r^2/2 - r^2\,\left(\arccos\left(1 - \frac{r - m\, z/2}{r}\right) - \left(1 - \frac{r - m\, z/2}{r}\right)\sqrt{\frac{2\, r - m\, z}{r} - \frac{(r - m\, z/2)^2}{r^2}}\right)}{m^2} & ,\ \text{if}\ \ \frac{z\, m}{2} \leq r \leq \frac{m}{z}, \qquad (11.14\text{b}) \\
\dfrac{\pi\, r^2/2 - r^2\,\left(\arccos\left(1 - \frac{r - m/z}{r}\right) - \left(1 - \frac{r - m/z}{r}\right)\sqrt{\frac{2\, r - 2\, m/z}{r} - \frac{(r - m/z)^2}{r^2}}\right)}{m^2} & ,\ \text{if}\ \ \frac{m}{z} \leq r \leq \frac{z\, m}{2}, \qquad (11.14\text{c}) \\
\dfrac{\pi\, r^2}{2\, m^2} - \dfrac{r^2}{m^2}\left(\arccos\left(1 - \dfrac{r - m\, z/2}{r}\right) - \left(1 - \dfrac{r - m\, z/2}{r}\right)\sqrt{\dfrac{2\, r - m\, z}{r} - \dfrac{(r - m\, z/2)^2}{r^2}}\right) & \qquad (11.14\text{d}) \\
\qquad - \dfrac{r^2}{m^2}\left(\arccos\left(1 - \dfrac{r - m/z}{r}\right) - \left(1 - \dfrac{r - m/z}{r}\right)\sqrt{\dfrac{2\, r - 2\, m/z}{r} - \dfrac{(r - m/z)^2}{r^2}}\right) & ,\ \text{if}\ \ r \geq \frac{m}{z},\ r \geq \frac{z\, m}{2}\ \text{and}\ r \leq \frac{\sqrt{5}\, m}{z}, \\
1 & ,\ \text{if}\ \ r \geq \frac{\sqrt{5}\, m}{z}\ . \qquad (11.14\text{e})
\end{cases}
$$

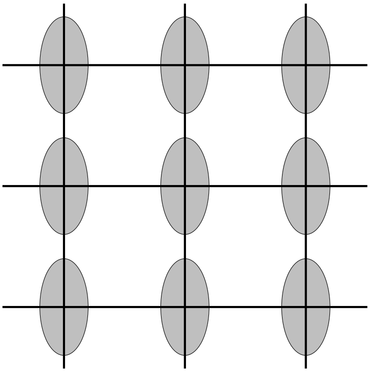


Figure 11.8: Square search grid with indication of the gray area enclosed around the search grid points by ellipses with length of long and short axis equal to r and $r/2$, respectively.

Next, $\mathcal{F}_{D_{\mathcal{G}}}$ is studied for different values of z and with $m = 1$, either directly making use of Eqs. (11.14a)–(11.14e) or graphically employing Figure 11.9. It is clear that an optimum is reached for $z = \sqrt{2}$ since larger or smaller values of z result in suboptimal solutions. This means that if the goal is to have a $\mathcal{F}_{D_{\mathcal{G}}}$ as large as possible, the optimal rectangular search grid when the contour lines are ellipses with axes parallel to the Cartesian coordinate axes, is a search grid where the ratio of the length of the rectangle sides is the same as the ratio of the length of the ellipse axes (in this case equal to two). The latter implies that circles are indeed a special case of ellipses since the ratio of the length of the axes of a circle is one which, when translated into the ratio of the lengths of the sides of the elementary parts of the search grid, results in squares, as shown earlier.

So, when the contour lines are ellipses, the naive square search grid can be improved by using a search grid where the elementary parts are rectangles with the ratio of the lengths of the rectangle sides equal to the ratio of the lengths of the ellipse axes. This information is contained in the Hessian $H(R)$. From $H(R)$, the eigenvalues can be calculated and as seen from Figure 11.10, the radii of the ellipsoidal contour are inversely proportional to the square roots of the eigenvalues of the matrix $H(R)$. So, when $H(R)$ is constructed, the eigenvalues of $H(R)$ contain the necessary information to improve the search grid.

Figure 11.8 shows ellipses with axes parallel to the axes of the search grid. However, more generally, there is no such limitation and ellipses can have

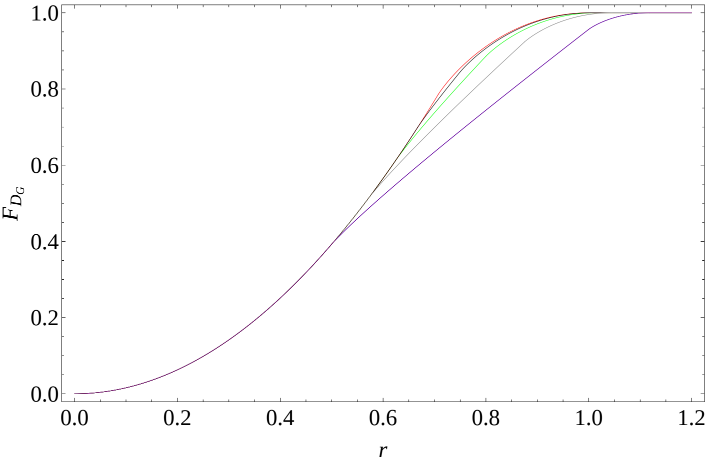


Figure 11.9: $\mathcal{F}_{D_{\mathcal{G}}}$ with $m = 1$ using Eqs. (11.14a)–(11.14e) for: (a) $z = 1$ (blue), (b) $z = 1.25$ (green), (c) $z = \sqrt{2}$ (red), (d) $z = 1.5$ (black), (e) $z = 1.75$ (gray) and (f) $z = 2.0$ (purple).

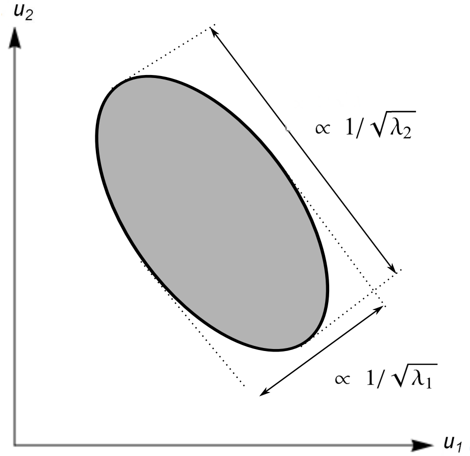


Figure 11.10: A contour line of $R(\boldsymbol{u})$. The radii of the ellipsoidal contour are inversely proportional to the square roots of the eigenvalues (λ_1 and λ_2) of the matrix $H(R)$.

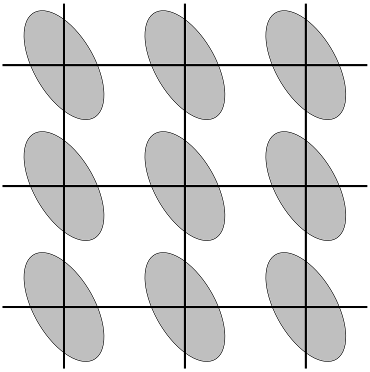


Figure 11.11: Square search grid with indication of the gray area enclosed around the search grid points by ellipses with length of long and short axis equal to r and $r/2$, respectively, and with axes not parallel to the search grid axes.

axes that are not parallel to the search grid axes, as depicted in Figure 11.11, i.e. when $2\sum_i x_i \neq 0$ (see supra). In this case, the eigenvectors of $H(R)$ are used to rotate the search grid axes. Let $A\,B\,A^\top$ denote the eigendecomposition of $H(R)$, where A is a square matrix of which the columns are eigenvectors of $H(R)$ and B is a diagonal matrix of which the diagonal elements are the eigenvalues of $H(R)$. Then, the rotated grid $\mathcal{G}'$ has search grid axes that are parallel to the ellipse axes:

$$\mathcal{G}' = \left\{\boldsymbol{u}^\top A \mid \boldsymbol{u} \in \mathcal{G}\right\}. \tag{11.15}$$

Subsequently, the eigenvalues can be employed to stretch the search grid like was done before.

11.1.2. Close-packing of equal spheres search grid

Although popular, a square search grid is not the most efficient one. A close-packing of equal spheres (CPES) provides the greatest fraction of space occupied by spheres that can be achieved by a regular grid arrangement [195]. Figure 11.12 shows this search grid with triangles with area $\sqrt{3}\,m^2$ as elementary parts. Since it is already stated that this configuration is the best attainable configuration when circles are concerned, in what follows this CPES grid is investigated for ellipsoidal contour lines.

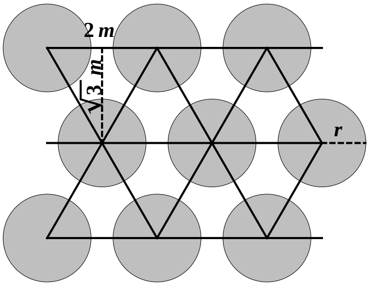


Figure 11.12: CPES search grid with indication of the gray area enclosed by circles with radius r around the search grid points.

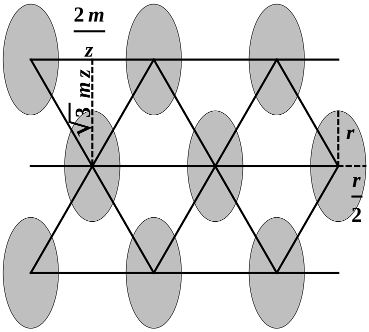


Figure 11.13: CPES search grid with indication of the gray area enclosed by ellipses with length of long and short axis equal to r and $r/2$, respectively.

Firstly, the ellipse axes are considered parallel to the Cartesian coordinate axes. Like before, ellipses where the ratio of the lengths of the ellipse axis parallel to the vertical grid axis to the one parallel to the horizontal grid axis is equal to two are considered. The area enclosed around the grid points by the ellipsoidal contours is represented in Figure 11.13. The length of the base of the equilateral triangular elementary cell is $2\,m/z$ and the height is $\sqrt{3}\,z\,m$, with $z \geq 1$. The ratio of the enclosed area to the total grid area in function of the threshold r can be determined from an elementary part of the grid.

As the explicit expression to calculate $\mathcal{F}_{D_{\mathcal{G}}}(r)$ would be very lengthy due to the number of possible scenarios by which three ellipses can overlap in a triangular elementary cell, $\mathcal{F}_{D_{\mathcal{G}}}(r)$ is determined by means of numerical simulations. A high number of points is selected uniformly in the elementary cell and subsequently the ratio of the number of points that are enclosed by an ellipsoidal contour to the total number of points of the elementary

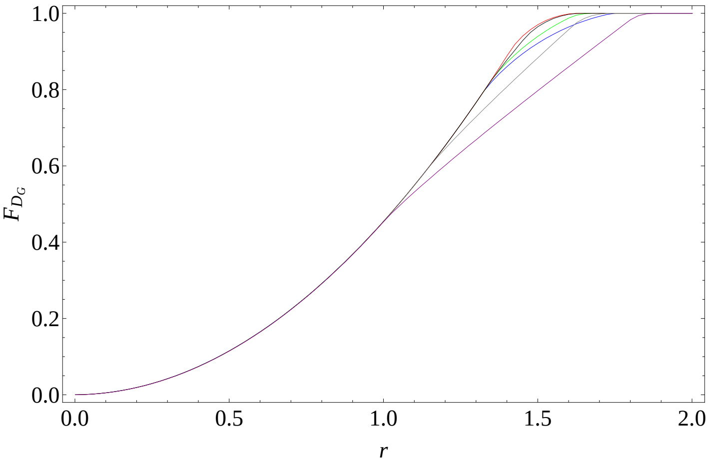
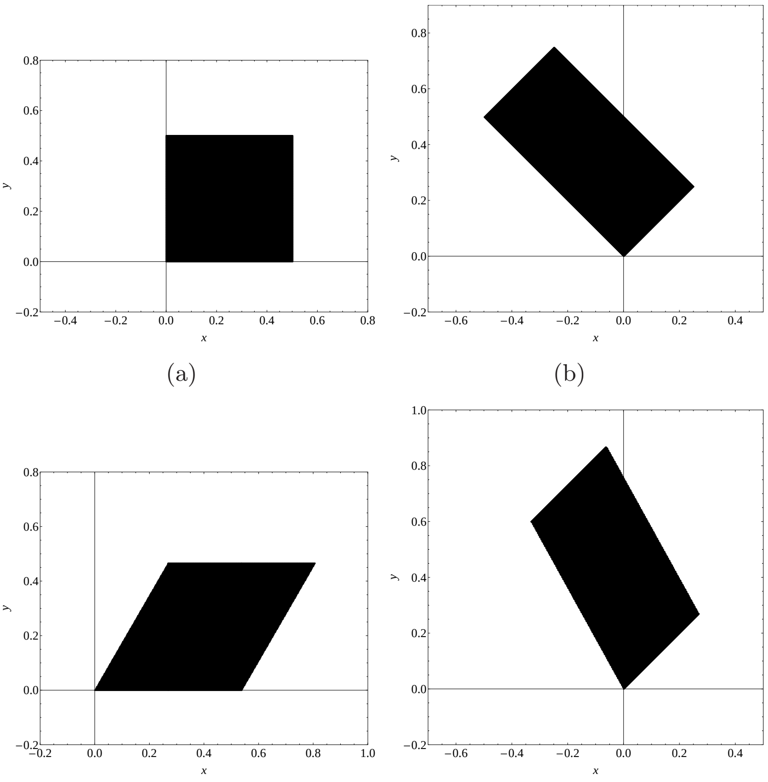


Figure 11.14: $\mathcal{F}_{D_\mathcal{G}}$ with $m = 1$ for ellipsoidal contours with axes parallel to the Cartesian coordinate axes and for a CPES search grid with base length $2\,m/z$ and height $\sqrt{3}\,z\,m$ for: (a) $z = 1$ (blue), (b) $z = 1.25$ (green), (c) $z = \sqrt{2}$ (red), (d) $z = 1.5$ (black), (e) $z = 1.75$ (gray) and (f) $z = 2.0$ (purple).

cell is determined. The result in function of the threshold value r for different values of z and with $m = 1$ is shown in Figure 11.14. Once more, the optimum is reached when $z = \sqrt{2}$, since larger or smaller values of z result in suboptimal solutions. So, it is concluded that the optimal CPES search grid is one where the ratio of the height of the triangular elementary cell and the length of base is equal to $\sqrt{3}/2$ times the ratio of the length of the ellipse axes, which can again be inferred from the eigenvalues of $H(R)$. Further, for ellipsoidal contours with axes that are not parallel to the search grid axes, again the eigenvectors of $H(R)$ are used to rotate the search grid axes after which the eigenvalues can again be employed to appropriately stretch the search grid.

11.1.3. Comparing search grids

In the following chapter, both the standard square search grid, because it is a popular choice and has the most straightforward implementation, as well as the standard CPES search grid, since it is an optimal regular grid arrangement, are used as starting points for the computational experiments. They are hereafter referred to as the naive square search grid ($\mathcal{G}_{\text{ns}}$) and the naive CPES search grid ($\mathcal{G}_{\text{ncp}}$). Their transformed counterparts using the information from $H(R)$ are denoted as $\mathcal{G}_{\text{ts}}$ and $\mathcal{G}_{\text{tcp}}$, respectively. Figure 11.16



(a) (b)

(c) (d)

Figure 11.15: Representation of an elementary part (with the same area in all cases) of the search grids for ellipsoidal contours whose axes form a 45° angle with the Cartesian coordinate axes and where the ratio of the length of the long axis to the length of the short axis is equal to two: (a) $\mathcal{G}_{\mathrm{ns}}$, (b) $\mathcal{G}_{\mathrm{ts}}$, (c) $\mathcal{G}_{\mathrm{ncp}}$ and (d) $\mathcal{G}_{\mathrm{tcp}}$.

shows $\mathcal{F}_{D_{\mathcal{G}}}$ for the four aforementioned grids (see Figures 11.15(a)–11.15(d)) for ellipsoidal contours whose axes form a 45° angle with the Cartesian coordinate axes and where the ratio of the length of the long axis to short axis is equal to two.

The elementary parts represented in Figures 11.15(a)–11.15(d) all have the same area, i.e. 0.25. Further, to enable the comparison of $\mathcal{G}_{\mathrm{ncp}}$ and $\mathcal{G}_{\mathrm{tcp}}$ with $\mathcal{G}_{\mathrm{ns}}$ and $\mathcal{G}_{\mathrm{ts}}$, two triangular elementary parts have to be considered together as such forming a parallelogram. This is explained by the fact that when an ellipsoidal contour is centered at each of the four corners of a (transformed) square grid, the four ellipsoidal parts (taking into account overlap) enclosed by the elementary part of the search grid add up to a

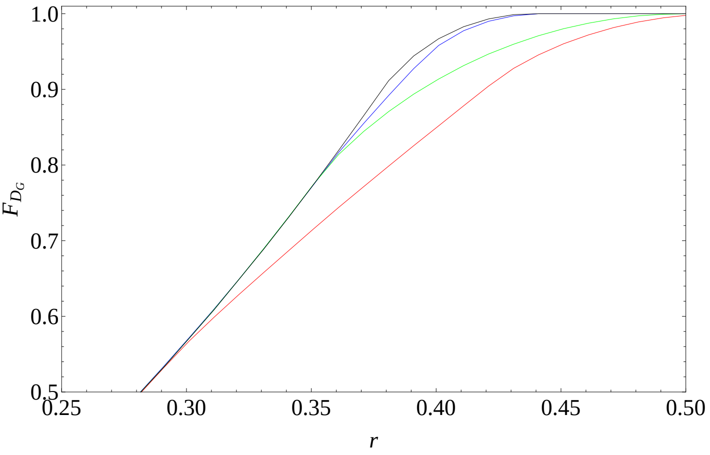


Figure 11.16: $\mathcal{F}_{D_{\mathcal{G}}}$ for ellipsoidal contours whose axes form a 45° angle with the Cartesian coordinate axes and where the ratio of the length of the long axis to the length of the short axis is equal to two for the search grids with elementary cells as shown in (a) Figure 11.15(a) (blue), (b) Figure 11.15(b) (green), (c) Figure 11.15(c) (red) and (d) Figure 11.15(d) (black).

complete ellipse. However, when the same is done for a (transformed) triangular grid, only half of an ellipse is enclosed. Therefore, two triangular elementary parts need to be considered together when comparing $\mathcal{G}_{\text{ncp}}$ and $\mathcal{G}_{\text{tcp}}$ with $\mathcal{G}_{\text{ns}}$ and $\mathcal{G}_{\text{ts}}$. From Figure 11.16, it can be seen that $\mathcal{G}_{\text{tcp}}$ performs better than $\mathcal{G}_{\text{ncp}}$ and more so, it performs the best out of the four search grids, which was expected. Nevertheless, $\mathcal{G}_{\text{tcp}}$ performs only slightly better than $\mathcal{G}_{\text{ns}}$, which outperforms $\mathcal{G}_{\text{ts}}$. This can be explained when looking at Figures 11.15(a) and 11.15(d), where it is clear that in this case $\mathcal{G}_{\text{ns}}$ shows a good resemblance to $\mathcal{G}_{\text{tcp}}$.

Finally, it is mentioned here that in many cases the stochastic dominance requirement of Eq. (11.4) is too strict for practical use. In order to still be able to make a statement about whether $\mathcal{G}_1$ performs better than $\mathcal{G}_2$, other measures such as the expected value of the random variables $D_{\mathcal{G}_1}$ and $D_{\mathcal{G}_2}$ have been adopted. In the following chapter, two such measures will be introduced and employed.

11.2. Sensitivity measures and the Hessian matrix

In this part, the information obtained from an SA will be used to improve the parameterization of models. The main advantages here are that the sensitivity information is already available through the model exploration and that this information is suitable for different parameter sets, since the sensitivity information does not depend on the experimental data. The use of SA information in improving parameter estimation is not new, although mostly it is used to guide (additional) experiments to gather data [84]. In the remainder of this section, it will be demonstrated how sensitivity measures obtained through a global SA can be used to improve the grid design for a grid search parameterization with a limited number of grid points.

Given a maximum SSE, denoted $\mathrm{SSE}_{\mathrm{acc}}$, that is acceptable for a good fit, the goal of this chapter is to select the grid points of the search grid in such a way that the probability of having at least one grid point within the area enclosed by the contour line corresponding to $\mathrm{SSE}_{\mathrm{acc}}$ is as high as possible. This can be done by exploiting the information from the contour plots. By projecting the ellipse on the vertical and horizontal axes in Figure 11.7, it is clear that the ellipse spans a wider range for u_2 than for u_1. Therefore, when selecting $\mathcal{N}$ points at which the model is to be evaluated, instead of selecting $\sqrt{\mathcal{N}}$ coordinate values equidistantly along both the u_1- and the u_2-axis and making all combinations with these coordinate values, it would be advantageous to select less coordinate values along the u_2-axis and more along the u_1-axis to increase the chance of finding at least one point within the contour line corresponding to $\mathrm{SSE}_{\mathrm{acc}}$.

Here, it is suggested that transforming a naive search grid $\mathcal{G}_{\mathrm{n}}$ by employing the eigenvalues and -vectors of $H(R)$ results on average in a better solution of the optimization problem. It will be shown that on average the SSE for the best solution found with the transformed search grid $\mathcal{G}_{\mathrm{t}}$ is smaller than the SSE of the best solution found with the naive search grid $\mathcal{G}_{\mathrm{n}}$.

It is noted that up to this point, there is no need for the information obtained through SA. However, in order to calculate the eigenvalues and -vectors using $H(R)$, it is a prerequisite that $f(\boldsymbol{x}, \boldsymbol{u})$ is a known expression. Yet, when working with black- and gray-box models (see Chapter 2) like was the case for the CA-based models developed in Part II, this is not the case, meaning that the eigenvalues and -vectors cannot be calculated via $H(R)$. Consequently, a modification of the approach is needed. Here, it is assumed that the interest lies in the neighborhood of the search space around the optimum $\boldsymbol{u}^*$ and that there it holds that $(f(\boldsymbol{x}, \boldsymbol{u}) - y_i) \approx 0$. In this setting, $H(R)$ approximately equals the Fisher information matrix (FIM):

$$\mathrm{FIM}(R) = \begin{pmatrix} \sum_i 2 \left(\frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_1} \right)^2 & \sum_i 2 \frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_1} \frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_2} \\ \sum_i 2 \frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_2} \frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_1} & \sum_i 2 \left(\frac{\partial f(\boldsymbol{x_i}, \boldsymbol{u})}{\partial u_2} \right)^2 \end{pmatrix}.$$

The Fisher information matrix is already being applied in the field of parameter estimation [196]. Here, the eigenvalues and -vectors of $\mathrm{FIM}(R)$ are used to determine $\mathcal{G}_{\mathrm{t}}^{\#}$ which is an approximation of $\mathcal{G}_{\mathrm{t}}$. Referring to Eq. (4.2), it can be observed that $\mathrm{FIM}(R)$ can be constructed using the SA information acquired during the model exploration as such avoiding the problem of not knowing the function explicitly. It will be shown in the next chapter that although $\mathcal{G}_{\mathrm{t}}^{\#}$ is an approximation of $\mathcal{G}_{\mathrm{t}}$, the use of the information it contains can lead to better grid search results.

A final remark is that when the model is nonlinear, the contour lines may have very complex shapes, deviating a lot from the ellipsoidal shape for which the approach is derived. Nevertheless, from Taylor's theorem it follows that in the neighborhood of the optimum the contour lines will approximate an ellipse such that the proposed approach of using $\mathcal{G}_{\mathrm{t}}$ or $\mathcal{G}_{\mathrm{t}}^{\#}$ rather than $\mathcal{G}_{\mathrm{n}}$ remains valid in this vicinity.

In the following chapter, the effectiveness of the proposed approach will firstly be demonstrated in the case of a linear model. In a second step, two well-known nonlinear models and a more complex gray-box CA-based model will be studied to verify whether the approach continues to produce better results for these models.

12 Computational experiments

12.1. Linear model

The effectiveness of the proposed approach is tested by constructing $\mathcal{G}_{\mathrm{t}}^{\#}$ and comparing the results obtained with those for $\mathcal{G}_{\mathrm{n}}$ for two different cases. The linear model introduced earlier (see Eq. (11.13)) is used here, where u_1 and u_2 are replaced with q_1 and q_2 in order to be in agreement with the notation introduced in Chapter 11, and with both belonging to $[0.1, 0.9]$. For a linear model it holds that $\mathrm{FIM}(R) = H(R)$ and therefore $\mathcal{G}_{\mathrm{t}}^{\#} = \mathcal{G}_{\mathrm{t}}$. In each of the following cases, a series of input values x_i is chosen and Eq. (11.13) is used to obtain the corresponding observations $y = f(x_i, \boldsymbol{q})$. Afterwards, $H(R)$ is calculated using this simulated dataset. Finally, the results obtained with both a naive square grid $\mathcal{G}_{\mathrm{ns}}$ and a naive CPES grid $\mathcal{G}_{\mathrm{ncp}}$ are compared with those obtained with their transformed variants $\mathcal{G}_{\mathrm{ts}}$ and $\mathcal{G}_{\mathrm{tcp}}$, respectively.

12.1.1. Case 1

In this first case, ellipsoidal contours whose axes are parallel to the Cartesian coordinate axes are considered. This implies that the series of inputs has to be centered around zero, i.e. $\sum\limits_i x_i = 0$. The chosen input series consists of 11 points, equidistantly chosen from the interval $[-1, 1]$. A possible contour plot for this case is shown in Figure 12.1.

Next, $\mathrm{FIM}(R)$ is calculated:

$$\mathrm{FIM}(R) = H(R) = \begin{pmatrix} 22 & 0 \\ 0 & 8.8 \end{pmatrix}.$$

At this point, it should be checked whether the ellipsoidal contours have the same shape and orientation throughout the selected search space. For the linear model this is always the case, however, if this were not the case, as will be seen with the nonlinear models, a subspace of the search space should be selected where the ellipsoidal contours (and consequently FIMs) are similar. Then, an average FIM can be calculated from different FIMs in

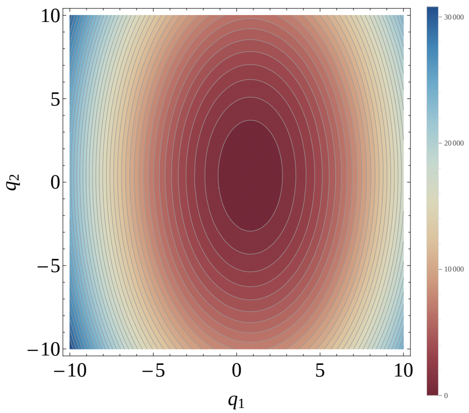
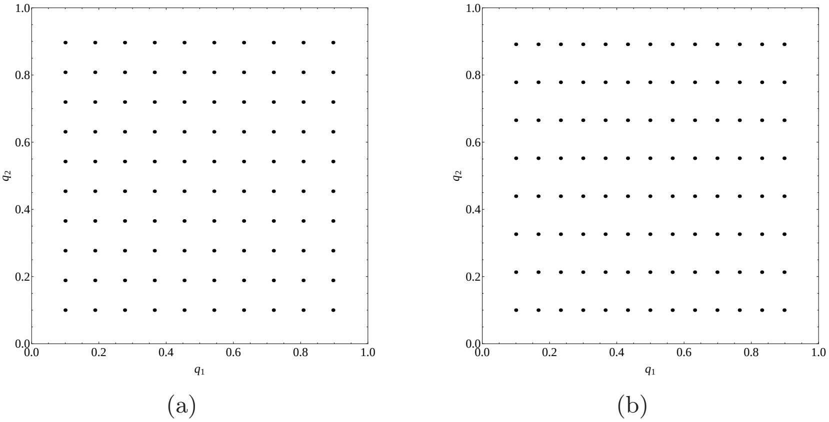


Figure 12.1: Contour plot of $R(\boldsymbol{q})$ for Case 1 of the linear model with $q_1^* = 0.82$ and $q_2^* = 0.39$.

this subspace, each with a different optimal parameter combination (q_1^*, q_2^*). However, for the linear case this is redundant and the eigenvalues and corresponding normalized eigenvectors are given by $\lambda_1 = 22$ and $v_1 = (-1, 0)^\top$ and $\lambda_2 = 8.8$ and $v_2 = (0, -1)^\top$. The eigenvectors suggest, in agreement with Figure 12.1 and with what was postulated, that a rotation of the naive grid in unnecessary, while the eigenvalues indicate a ratio of the ellipse axes β of $\sqrt{22/8.8} = 1.58$.

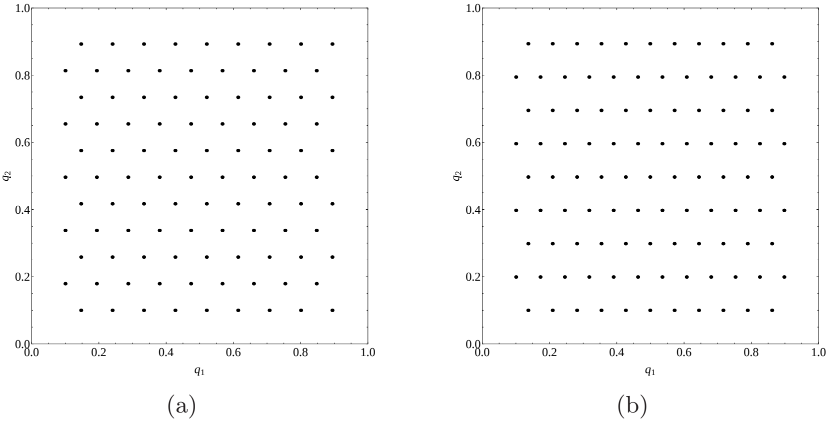
Figures 12.2(a) and 12.2(b) and Figures 12.3(a) and 12.3(b) illustrate how β is used in this case to transform $\mathcal{G}_{\text{ns}}$ into $\mathcal{G}_{\text{ts}}$ and $\mathcal{G}_{\text{ncp}}$ into $\mathcal{G}_{\text{tcp}}$, respectively. It is mentioned here that always getting the exact same number of grid points in both naive search grids and their transformed variants using β is impossible. A small deviance in the number of grid points and/or β is sometimes necessary in order to compare the different search grids. The grids $\mathcal{G}_{\text{ns}}$, $\mathcal{G}_{\text{ts}}$, $\mathcal{G}_{\text{ncp}}$ and $\mathcal{G}_{\text{tcp}}$ contain 100, 104, 99 and 103 grid points, respectively. Further, for the transformation of $\mathcal{G}_{\text{ns}}$ to $\mathcal{G}_{\text{ts}}$ a value of $\beta = 1.70$ was employed, instead of 1.58.

Table 12.1 shows the numerical results from the comparison of the different approaches for Case 1. The results are obtained by selecting a random $\boldsymbol{q}^*$ 2000 times and calculating the corresponding observed values for the selected series of inputs, after which the different search grids are used to perform the parameter estimation. The minimal SSEs found with the different search grids are then compared. The improvement of the proposed approach is quantified in two ways.



(a) (b)

Figure 12.2: Transformation of (a) a square search grid $\mathcal{G}_{\mathrm{ns}}$ with 100 grid points into (b) a transformed grid $\mathcal{G}_{\mathrm{ts}}$ with 104 grid points using $\beta = 1.70$.



(a) (b)

Figure 12.3: Transformation of (a) a CPES search grid $\mathcal{G}_{\mathrm{ncp}}$ with 99 grid points into (b) a transformed grid $\mathcal{G}_{\mathrm{tcp}}$ with 103 grid points using $\beta = 1.58$.

$\mathcal{G}_1$	$\mathcal{G}_2$	$\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$	$\Theta_{\mathcal{G}_1,\mathcal{G}_2}$
$\mathcal{G}_{\mathrm{ts}}$	$\mathcal{G}_{\mathrm{ns}}$	0.552	1.217
$\mathcal{G}_{\mathrm{tcp}}$	$\mathcal{G}_{\mathrm{ncp}}$	0.514	1.142
$\mathcal{G}_{\mathrm{ncp}}$	$\mathcal{G}_{\mathrm{ns}}$	0.548	1.244
$\mathcal{G}_{\mathrm{tcp}}$	$\mathcal{G}_{\mathrm{ts}}$	0.534	1.104

Table 12.1: Numerical results for Case 1 of the linear model.

The first way is by means of the measure $\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$ which expresses the fraction of the simulated cases in which a lower SSE is found with $\mathcal{G}_1$ compared with $\mathcal{G}_2$:

$$\Gamma_{\mathcal{G}_1,\mathcal{G}_2} = \frac{1}{2000} \sum_{i=1}^{2000} I\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} < \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right), \tag{12.1}$$

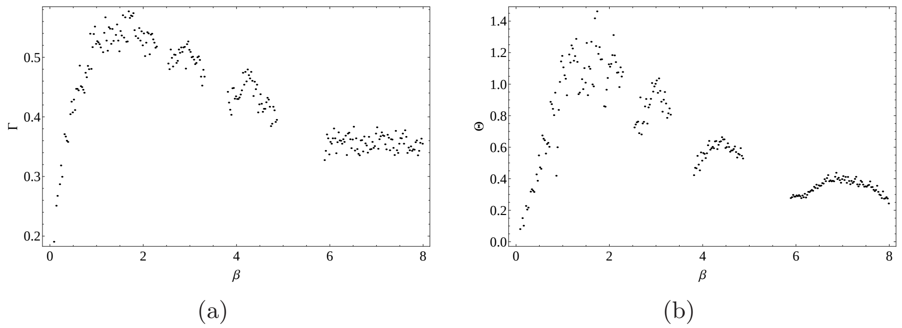
with I the indicator function which returns the value 1 for all elements where $\mathrm{SSE}^{(i)}_{\mathcal{G}_1} < \mathrm{SSE}^{(i)}_{\mathcal{G}_2}$ is true and the value 0 for all other elements and $\mathrm{SSE}^{(i)}_{\mathcal{G}}$ the lowest SSE obtained with grid $\mathcal{G}$ for the ith repetition. Clearly, a value of $\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$ greater than 0.5 is indicative for the superiority of $\mathcal{G}_1$ over $\mathcal{G}_2$. Note that Γ is additively reciprocal, i.e. $\Gamma_{\mathcal{G}_1,\mathcal{G}_2} + \Gamma_{\mathcal{G}_2,\mathcal{G}_1} = 1$.

Secondly, the measure $\Theta_{\mathcal{G}_1,\mathcal{G}_2}$ represents the ratio of the absolute average difference in SSE between $\mathcal{G}_1$ and $\mathcal{G}_2$ when $\mathcal{G}_1$ performs better than $\mathcal{G}_2$ to the absolute average difference when $\mathcal{G}_2$ performs better than $\mathcal{G}_1$:

$$\Theta_{\mathcal{G}_1,\mathcal{G}_2} = \frac{\sum_{i=1}^{2000} \left(I\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} < \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right)\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_2} - \mathrm{SSE}^{(i)}_{\mathcal{G}_1}\right)\right) \Big/ \sum_{i=1}^{2000} I\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} < \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right)}{\sum_{i=1}^{2000} \left(I\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} > \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right)\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} - \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right)\right) \Big/ \sum_{i=1}^{2000} I\left(\mathrm{SSE}^{(i)}_{\mathcal{G}_1} > \mathrm{SSE}^{(i)}_{\mathcal{G}_2}\right)}. \tag{12.2}$$

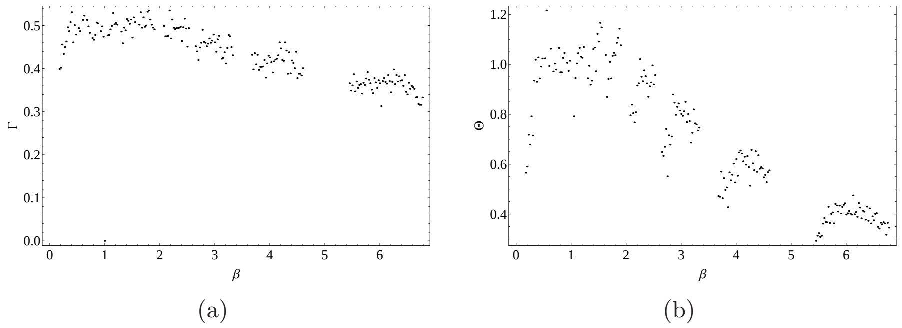
Clearly, a value of $\Theta_{\mathcal{G}_1,\mathcal{G}_2}$ greater than 1 is indicative for the superiority of $\mathcal{G}_1$ over $\mathcal{G}_2$. Note that Θ is multiplicatively reciprocal, i.e. $\Theta_{\mathcal{G}_1,\mathcal{G}_2}\,\Theta_{\mathcal{G}_2,\mathcal{G}_1} = 1$.

From Table 12.1 it is clear that transforming the naive grids leads to better results in both cases. Further, an improvement can also be made by simply selecting a CPES grid compared with a naive square grid. The results in this specific case do not amount to a large profit for switching from a naive to a transformed grid. However, this is not inherent to the method, but



(a) (b)

Figure 12.4: Influence of β on the transformation of the square search grid: (a) $\Gamma_{\mathcal{G}_{\text{ts}},\mathcal{G}_{\text{ns}}}$ and (b) $\Theta_{\mathcal{G}_{\text{ts}},\mathcal{G}_{\text{ns}}}$, both in function of β for Case 1 of the linear model.



(a) (b)

Figure 12.5: Influence of β on the transformation of the naive CPES grid: (a) $\Gamma_{\mathcal{G}_{\text{tcp}},\mathcal{G}_{\text{ncp}}}$ and (b) $\Theta_{\mathcal{G}_{\text{tcp}},\mathcal{G}_{\text{ncp}}}$, both in function of β for Case 1 of the linear model.

rather due to the relative small difference in parameter sensitivity which is reflected by the low value of β, i.e. 1.58. Larger values for β in this case would result in larger profit from using the proposed approach.

Figures 12.4(a) and 12.4(b) show $\Gamma_{\mathcal{G}_{\text{ts}},\mathcal{G}_{\text{ns}}}$ and $\Theta_{\mathcal{G}_{\text{ts}},\mathcal{G}_{\text{ns}}}$ in function of β, respectively. It is clear from these figures that not for every value of β a result for Γ and Θ is available. This can be explained by the fact that when the number of grid points of the transformed grid deviated more than 5% (both higher and lower) from the total number of grid points of the naive grid, this β was not taken into consideration. In both cases, there exists a maximum around $\beta = 1.58$ which reconfirms the effectiveness of the proposed procedure. Similar results are obtained when investigating the influence of β on the comparison between $\mathcal{G}_{\text{ncp}}$ and $\mathcal{G}_{\text{tcp}}$ (see Figures 12.5(a) and 12.5(b)).

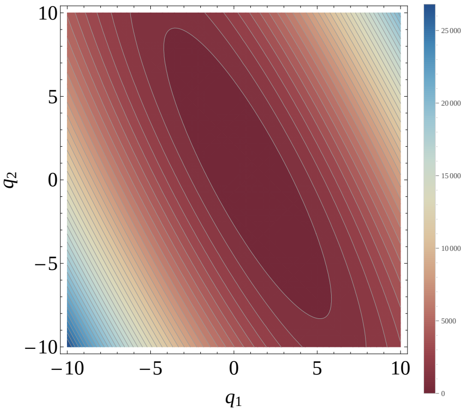


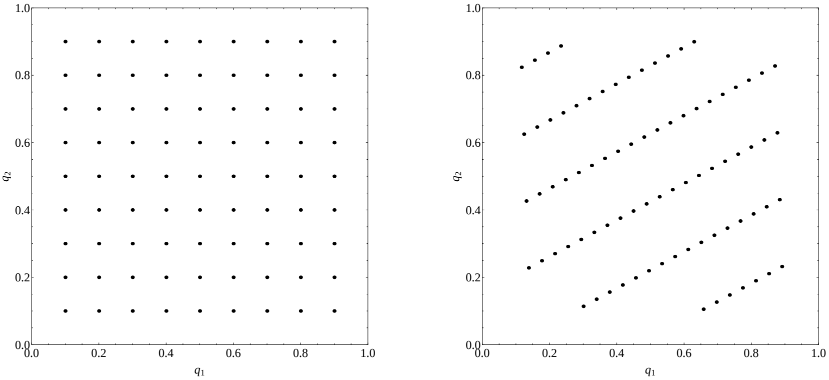
Figure 12.6: Contour plot of $R(\boldsymbol{q})$ for Case 2 of the linear model with $q_1^* = 0.82$ and $q_2^* = 0.39$.

12.1.2. Case 2

The second case of the linear model addresses ellipsoidal contours with axes that are not parallel to the search grid axes. The series of 11 inputs is chosen equidistantly from the interval $[0, 1]$. Figure 12.6 shows a possible contour plot for this case.

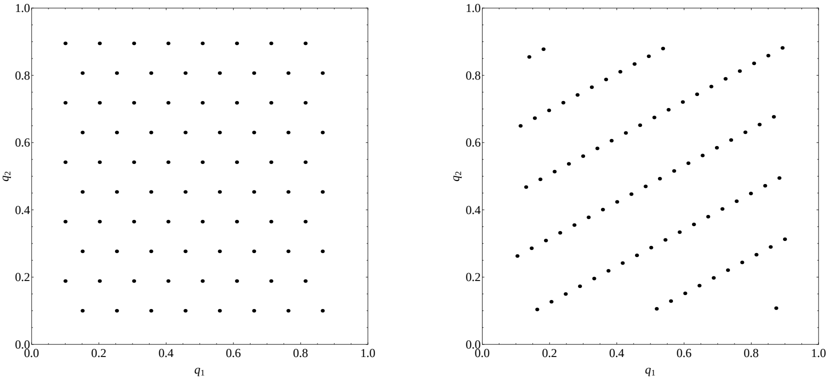
The eigenvalues and corresponding normalized eigenvectors for this case are given by $\lambda_1 = 27.97$ and $v_1 = (-0.88, -0.48)^\top$ and $\lambda_2 = 1.73$ and $v_2 = (0.48, -0.88)^\top$. As in the previous case, these eigenvalues and -vectors are employed to attempt to improve search grid results. From the eigenvalues, a value of $\beta = 4.02$ is obtained. The number of grid points for $\mathcal{G}_{\text{ns}}$, $\mathcal{G}_{\text{ts}}$, $\mathcal{G}_{\text{ncp}}$ and $\mathcal{G}_{\text{tcp}}$ is 81, 81, 80 and 80, respectively, and in both transformations the calculated β is used. Figures 12.7(a) and 12.7(b) and Figures 12.8(a) and 12.8(b) illustrate how β and the eigenvectors are used in this case to transform $\mathcal{G}_{\text{ns}}$ into $\mathcal{G}_{\text{ts}}$ and $\mathcal{G}_{\text{ncp}}$ into $\mathcal{G}_{\text{tcp}}$, respectively.

Given the numerical results in Table 12.2, it can be concluded that equally good results are obtained with $\mathcal{G}_{\text{tcp}}$ and $\mathcal{G}_{\text{ns}}$. It is clear that transforming $\mathcal{G}_{\text{ns}}$ to $\mathcal{G}_{\text{ts}}$ is not fruitful. The reason is that $\mathcal{G}_{\text{ns}}$ resembles the (transformed) close-packing grid better than its transformed counterpart, therefore nullifying the effect of employing the information from $H(R)$ to improve results. Still more, the largest gain comes from simply using $\mathcal{G}_{\text{ns}}$ instead of $\mathcal{G}_{\text{ncp}}$. For this improvement in performance, no extra calculations need to be performed since it only depends on the choice of basic search grid. Nev-



(a) (b)

Figure 12.7: Transformation of (a) a square search grid $\mathcal{G}_{\mathrm{ns}}$ with 81 grid points into (b) a transformed grid $\mathcal{G}_{\mathrm{ts}}$ with 81 grid points using $\beta = 4.02$ and rotation via the normalized eigenvectors v_1 and v_2.



(a) (b)

Figure 12.8: Transformation of (a) a CPES search grid $\mathcal{G}_{\mathrm{ncp}}$ with 80 grid points into (b) a transformed grid $\mathcal{G}_{\mathrm{tcp}}$ with 80 grid points using $\beta = 4.02$ and rotation via the normalized eigenvectors v_1 and v_2.

$\mathcal{G}_1$	$\mathcal{G}_2$	$\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$	$\Theta_{\mathcal{G}_1,\mathcal{G}_2}$
$\mathcal{G}_{\text{ts}}$	$\mathcal{G}_{\text{ns}}$	0.574	0.838
$\mathcal{G}_{\text{tcp}}$	$\mathcal{G}_{\text{ncp}}$	0.639	2.242
$\mathcal{G}_{\text{ncp}}$	$\mathcal{G}_{\text{ns}}$	0.381	0.427
$\mathcal{G}_{\text{tcp}}$	$\mathcal{G}_{\text{ts}}$	0.499	1.163
$\mathcal{G}_{\text{tcp}}$	$\mathcal{G}_{\text{ns}}$	0.566	0.959

Table 12.2: Numerical results for Case 2 of the linear model.

ertheless, stretching and rotating $\mathcal{G}_{\text{ncp}}$ also results in an equally good result which comes, if the SA information is already available from the model exploration, at no extra cost in terms of model evaluations.

12.2. Nonlinear models

In the remainder of this chapter, it will be investigated whether the conclusions drawn before uphold when FIM is only an approximation for H and when the contour lines are not the same throughout the search space. For this purpose, two well-known nonlinear models, the SIRD epidemic model [197] and the Predator-Prey model [4], and the CA-based model for oil migration (see Chapter 8) are used as an illustration.

12.2.1. SIRD epidemic model

The study of the transmission of communicable diseases through individuals can be done using an epidemic model. In this section, the basic SIRD model is employed [197], although many more (complex) variants of this epidemic model exist. The change in function of time in the population of susceptible, infected, dead and resistant people is described by:

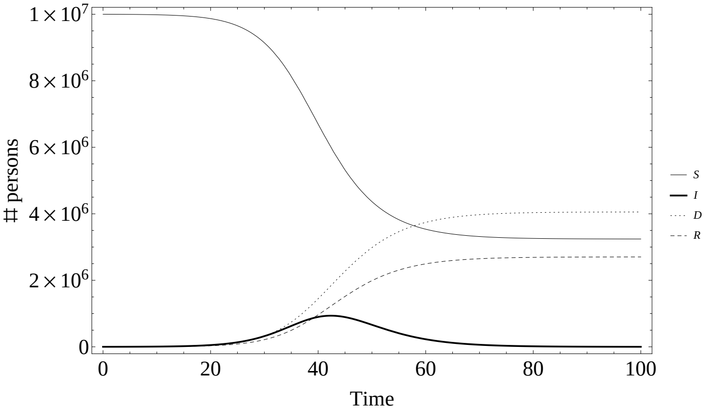


Figure 12.9: Evolution in time of the population during an epidemic (see (12.3)) with $S(0) = 10,000,000$, $I(0) = 1000$ and $R(0) = D(0) = 0$ and $a = 0.05$, $b = 0.000001$, $c = 0.3$ and $w = 0.6$.

$$\begin{cases} \dfrac{dS}{dt} = -a\,b\,I\,S\,, \\ \dfrac{dI}{dt} = a\,b\,I\,S - c\,I\,, \\ \dfrac{dD}{dt} = w\,c\,I\,, \\ \dfrac{dR}{dt} = (1-w)\,c\,I\,, \end{cases} \tag{12.3}$$

with S, I, D and R the number of susceptible, infected, dead and resistant individuals, respectively, a $[\frac{\text{individual}}{\text{contact}}]$ the probability of getting infected upon contact with an infected individual, b $[\frac{\text{contact}}{\text{individual}^2\,\text{day}}]$ the interaction probability of two individuals from the population, c $[\frac{1}{\text{day}}]$ the fraction per day of infected persons that is no longer infected and w [-] the fraction of individuals that dies in consequence of the infection. The solution to this system of nonlinear differential equations with initial conditions $S(0) = 10000000$, $I(0) = 1000$ and $R(0) = D(0) = 0$ and $a = 0.05$, $b = 0.000001$, $c = 0.3$ and $w = 0.6$ is shown in Figure 12.9.

For reasons of simplicity, hereafter only I will be used as model output and b and w are assigned a constant value, i.e. 0.000001 and 0.6 respectively, such that only two model parameters, a and c remain. After some

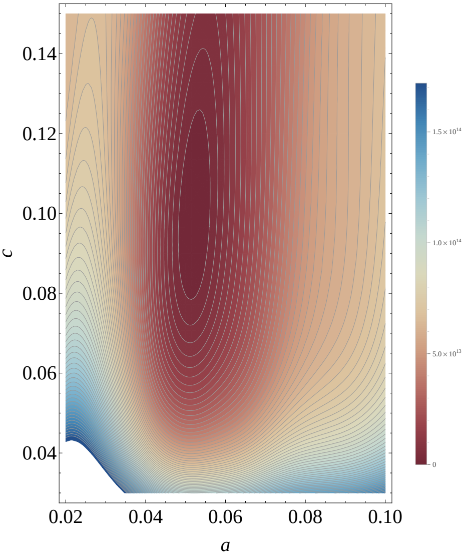


Figure 12.10: Contour plot of $R(a, c)$ for the SIRD epidemic model with $a^* = 0.052$ and $c^* = 0.098$.

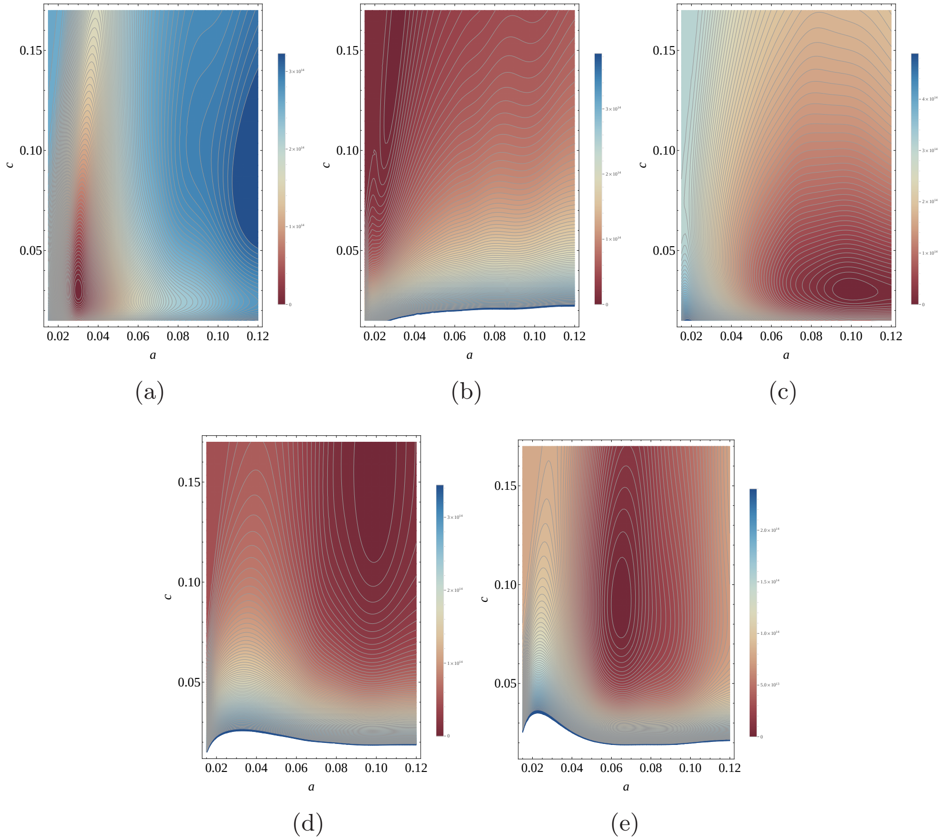
model exploration, it can be concluded that plausible results for I are obtained for combinations of a and c where $a \in [0.02, 0.1]$ and $c \in [0.03, 0.15]$. The selected series of inputs contains 21 points, equidistantly chosen from the interval $[0, 100]$. A possible contour plot for the objective function is depicted in Figure 12.10.

Constructing $H(R)$ and $\text{FIM}(R)$ as explained before with $a^* = 0.052$ and $c^* = 0.098$ gives

$$H(R) = \begin{pmatrix} 3.202 \times 10^{17} & -1.337 \times 10^{16} \\ -1.337 \times 10^{16} & 9.544 \times 10^{15} \end{pmatrix},$$

$$\text{FIM}(R) = \begin{pmatrix} 3.202 \times 10^{17} & -1.337 \times 10^{16} \\ -1.337 \times 10^{16} & 9.544 \times 10^{15} \end{pmatrix}.$$

Comparing both, it can be seen that although $\text{FIM}(R)$ is constructed using SA information and therefore is an approximation of $H(R)$ that, to the accuracy presented here, it is equal to $H(R)$. Figures 12.11(a)–12.11(e) show that the ellipsoidal contours have varying shapes and orientation across the search space. Especially for large values of a in combination with small values of c, e.g. Figure 12.11(c), the orientation of the ellipsoidal contours is clearly different. Therefore, a subspace of the original search space is se-



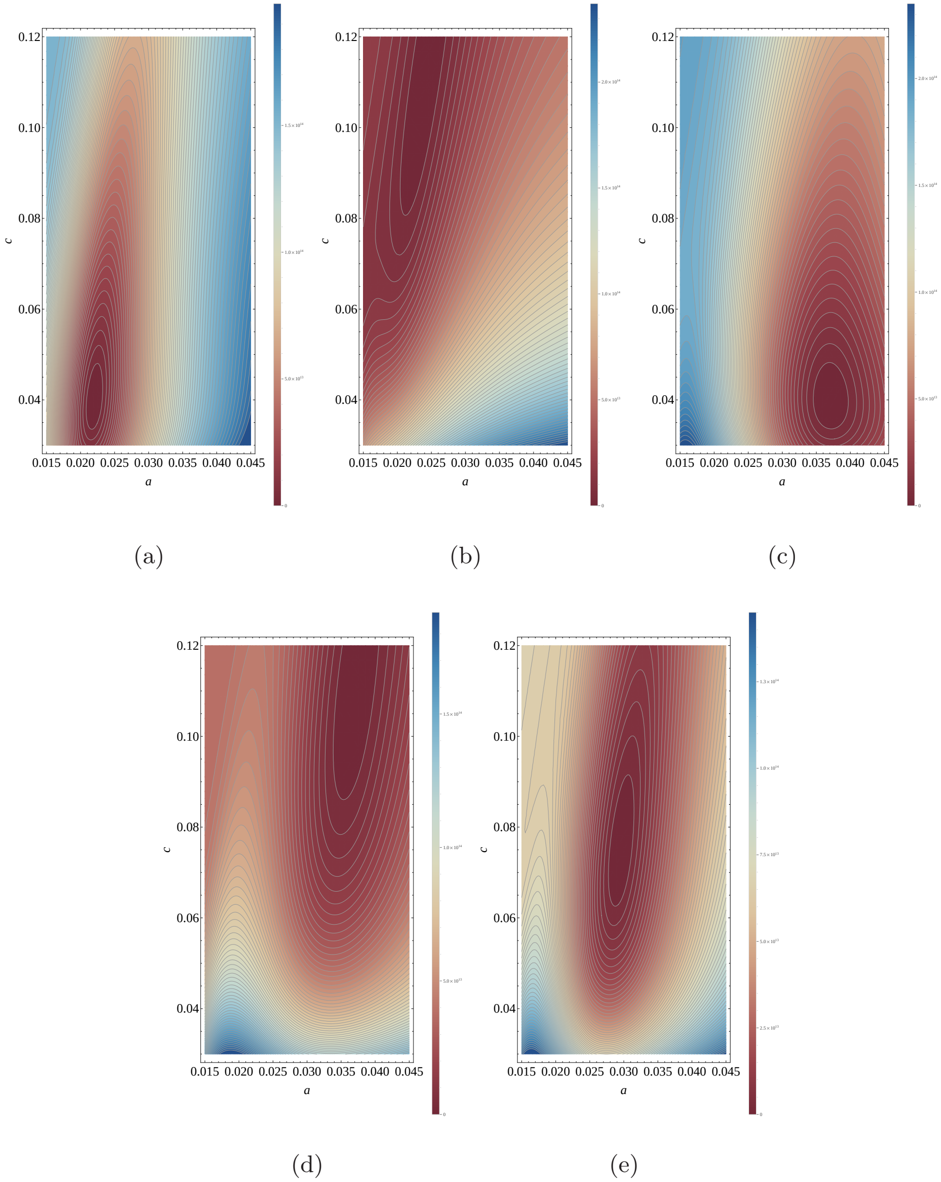
(a) (b) (c)

(d) (e)

Figure 12.11: Contour plots of $R(a, c)$ for the SIRD epidemic model with (a) $a^* = c^* = 0.03$, (b) $a^* = 0.03$ and $c^* = 0.15$, (c) $a^* = 0.1$ and $c^* = 0.03$, (d) $a^* = 0.1$ and $c^* = 0.15$ and (e) $a^* = 0.065$ and $c^* = 0.09$.

lected wherein the ellipsoidal contours are more similar. Figures 12.12(a)–12.12(e) show the contour plots for different couples (a^*, c^*) in the subspace where $a^* \in [0.022, 0.037]$ and $c^* \in [0.04, 0.11]$.

FIM(R) is determined for 25 different couples (a^*, c^*) sampled randomly from the subspace $[0.022, 0.037] \times [0.04, 0.11]$ after which an average FIM(R) is calculated. For this average FIM(R), the eigenvalues and corresponding normalized eigenvectors are given by $\lambda_1 = 1.684 \times 10^{18}$ and $\lambda_2 = 4.017 \times 10^{16}$ and $v_1 = (-0.998, 0.061)^\top$ and $v_2 = (-0.061, -0.998)^\top$, respectively. From λ_1 and λ_2 it follows that $\beta = 6.48$. The eigenvectors, together with Figures 12.12(a)–12.12(e), indicate that this case is similar to Case 1 of the linear model. In what follows, the same experimental procedure as for Case 1 of the linear model is followed after which the conclusions of both



(a) (b) (c)

(d) (e)

Figure 12.12: Contour plots of $R(a, c)$ for the SIRD epidemic model with (a) $a^* = 0.022$ and $c^* = 0.04$, (b) $a^* = 0.022$ and $c^* = 0.11$, (c) $a^* = 0.037$ and $c^* = 0.04$, (d) $a^* = 0.037$ and $c^* = 0.11$ and (e) $a^* = 0.0295$ and $c^* = 0.075$.

$\mathcal{G}_1$	$\mathcal{G}_2$	$\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$	$\Theta_{\mathcal{G}_1,\mathcal{G}_2}$
$\mathcal{G}^{\#}_{\mathrm{ts}}$	$\mathcal{G}_{\mathrm{ns}}$	0.732	6.711
$\mathcal{G}^{\#}_{\mathrm{tcp}}$	$\mathcal{G}_{\mathrm{ncp}}$	0.469	1.063
$\mathcal{G}_{\mathrm{ncp}}$	$\mathcal{G}_{\mathrm{ns}}$	0.736	5.000
$\mathcal{G}^{\#}_{\mathrm{tcp}}$	$\mathcal{G}^{\#}_{\mathrm{ts}}$	0.494	0.920

Table 12.3: Numerical results for the nonlinear SIRD epidemic model.

are compared in order to verify the effectiveness of the proposed approach. The number of grid points for $\mathcal{G}_{\mathrm{ns}}$, $\mathcal{G}^{\#}_{\mathrm{ts}}$, $\mathcal{G}_{\mathrm{ncp}}$ and $\mathcal{G}^{\#}_{\mathrm{tcp}}$ is 120, 121, 120 and 120, respectively and in the transformation of $\mathcal{G}_{\mathrm{ncp}}$ to $\mathcal{G}^{\#}_{\mathrm{tcp}}$, $\beta = 6.55$ is employed.

Table 12.3 confirms the conclusions found for Case 1 of the linear model, indicating that the transformed grids outperform the naive grids. The results in this case are more pronounced than for the linear model since β is larger. Further, it is noticed that also here the choice of a CPES search grid over a naive square grid greatly improves the results. Finally, it can be seen that the results obtained with $\mathcal{G}^{\#}_{\mathrm{ts}}$ approximate the results obtained with $\mathcal{G}^{\#}_{\mathrm{tcp}}$ in this specific case.

12.2.2. Predator-Prey model

The Predator-Prey equations are a pair of first-order, nonlinear, differential equations that are often employed to describe the dynamics of biological systems in which two species, one predator and one prey, interact [4]. The evolution in time of these two species is described by:

$$\begin{cases} \dfrac{dJ}{dt} = g\, J - d\, J\, P\,, \\[2ex] \dfrac{dP}{dt} = -s\, P + e\, J\, P\,, \end{cases} \tag{12.4}$$

with J the number of preys, P the number of predators, g the growth rate of the preys in absence of predators, d the mortality rate of preys due to predation, s the mortality rate of predators in absence of preys and e the growth rate of predators due to predation of preys. Figure 12.13 shows the

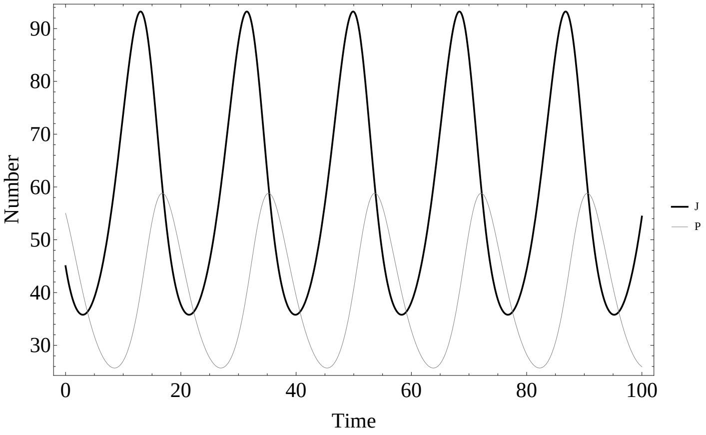


Figure 12.13: Evolution in time of the number of predators and preys (see (12.4)) with $J(0) = 45$, $P(0) = 55$, $g = 0.4$, $d = 0.01$, $s = 0.3$ and $e = 0.005$.

solution of System (12.4) for initial conditions $J(0) = 45$ and $P(0) = 55$ and $g = 0.4$, $d = 0.01$, $s = 0.3$ and $e = 0.005$.

Like with the SIRD epidemic model, only one output is considered, in this case J, and two model parameters. A constant value of 0.01 and 0.005 is assigned to d and e, respectively, such that only g and s remain. After some model exploration, it is clear that plausible results for J are obtained when both g and s belong to $[0.3, 0.5]$. The selected series of inputs contains 21 points, equidistantly chosen from the interval $[0, 100]$. The contour plot for the objective function pertaining to $g^* = 0.31$ and $s^* = 0.47$ is given in Figure 12.14.

$H(R)$ and $\mathrm{FIM}(R)$ are calculated with $g^* = 0.31$ and $s^* = 0.47$ as

$$H(R) = \begin{pmatrix} 8.255 \times 10^7 & 4.880 \times 10^7 \\ 4.880 \times 10^7 & 3.226 \times 10^7 \end{pmatrix},$$

$$\mathrm{FIM}(R) = \begin{pmatrix} 8.253 \times 10^7 & 4.877 \times 10^7 \\ 4.877 \times 10^7 & 3.225 \times 10^7 \end{pmatrix}.$$

It can be seen that although $H(R)$ and $\mathrm{FIM}(R)$ are not the same to the accuracy presented here, they are still very similar. Figures 12.15(a)–12.15(e) show that the ellipsoidal contours again have varying shapes and orientation across the search space, but that this variation is acceptable within the goals of this research, such that an average $\mathrm{FIM}(R)$ can be calculated immediately.

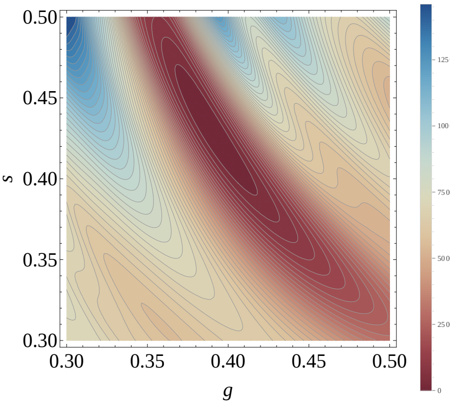
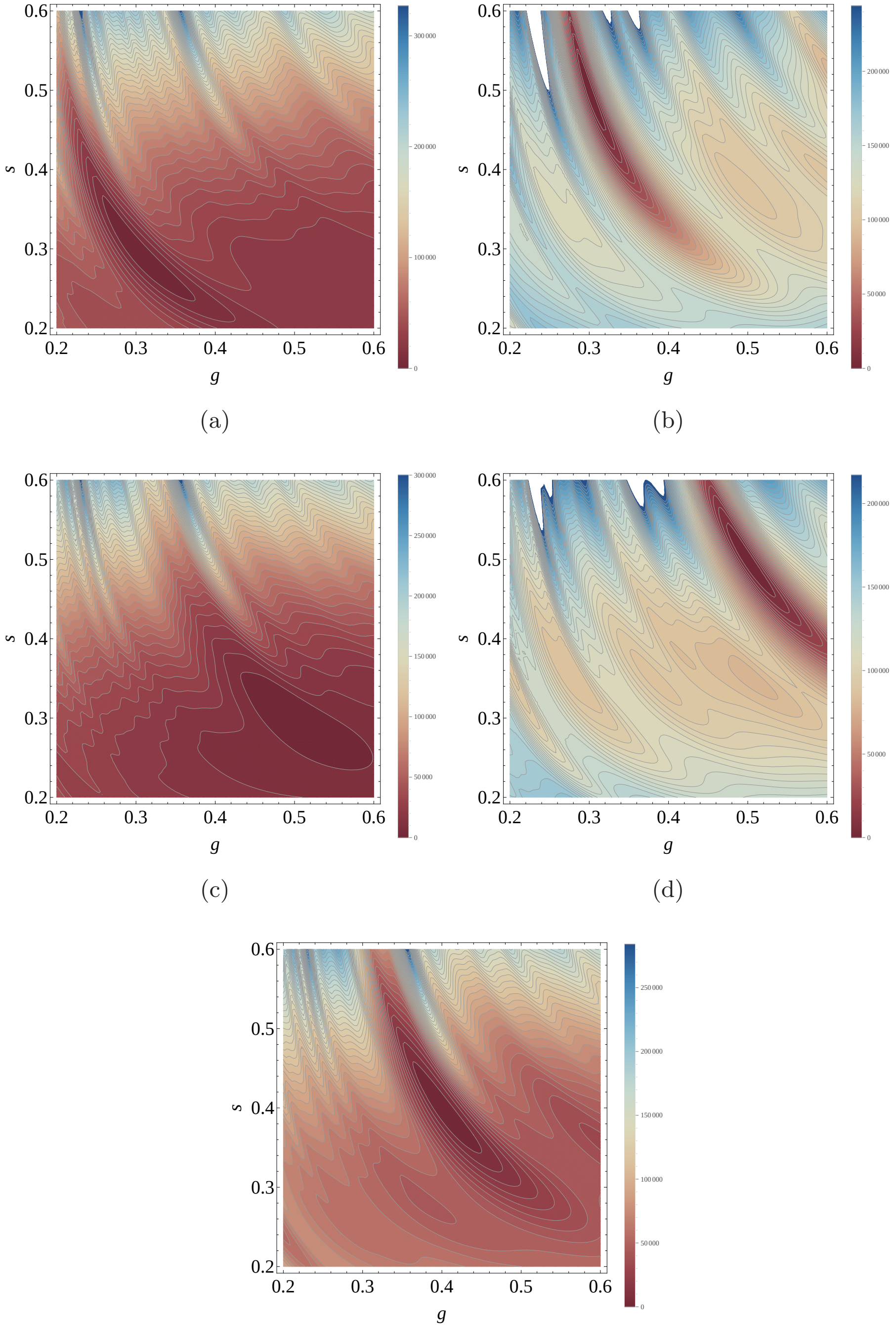


Figure 12.14: Contour plot of $R(g, s)$ for the Predator-Prey model with $g^* = 0.31$ and $s^* = 0.47$.

FIM(R) corresponding to the search space is taken as the average FIM(R) of 25 FIMs obtained using 25 different couples (g^*, s^*) randomly taken from the search space. For this average FIM(R), the eigenvalues and corresponding normalized eigenvectors are given by $\lambda_1 = 1.632 \times 10^8$ and $v_1 = (-0.483, -0.220)^\top$ and $\lambda_2 = 2.261 \times 10^6$ and $v_2 = (-0.220, 0.483)^\top$. Using λ_1 and λ_2, β can be calculated as 8.50. The eigenvectors, together with Figures 12.15(a)–12.15(e), indicate that this case is similar to Case 2 of the linear model. It is verified whether the same conclusions as before uphold for the nonlinear model. The number of grid points for $\mathcal{G}_{\text{ns}}$, $\mathcal{G}^{\#}_{\text{ts}}$, $\mathcal{G}_{\text{ncp}}$ and $\mathcal{G}^{\#}_{\text{tcp}}$ is 100, 100, 99 and 101, respectively and in both transformations $\beta = 8.50$ is employed.

The same conclusions can be inferred from Table 12.4 as from Table 12.2. The biggest improvement can be obtained when switching from $\mathcal{G}_{\text{ncp}}$ to $\mathcal{G}_{\text{ns}}$. In this case, it is again apparent that simply using $\mathcal{G}_{\text{ns}}$ is a good option, due to its resemblance to a CPES search grid relatively seen to the ellipsoidal contours such as the ones depicted in Figure 12.14. It can be seen that transforming $\mathcal{G}_{\text{ns}}$ to obtain $\mathcal{G}^{\#}_{\text{ts}}$ does not lead to an improvement. Moreover, in the case of the Predator-Prey model, $\mathcal{G}_{\text{ns}}$ performs equally good as the theoretically best grid, i.e. $\mathcal{G}^{\#}_{\text{tcp}}$.



(a)

(b)

(c)

(d)

(e)

Figure 12.15: Contour plots of $R(g,s)$ for the Predator-Prey model with (a) $g^* = s^* = 0.3$, (b) $g^* = 0.3$ and $g^* = 0.5$, (c) $g^* = 0.5$ and $g^* = 0.3$, (d) $g^* = s^* = 0.5$ and (e) $g^* = s^* = 0.4$.

$\mathcal{G}_1$	$\mathcal{G}_2$	$\Gamma_{\mathcal{G}_1,\mathcal{G}_2}$	$\Theta_{\mathcal{G}_1,\mathcal{G}_2}$
$\mathcal{G}_{\text{ts}}^{\#}$	$\mathcal{G}_{\text{ns}}$	0.572	0.737
$\mathcal{G}_{\text{tcp}}^{\#}$	$\mathcal{G}_{\text{ncp}}$	0.643	1.739
$\mathcal{G}_{\text{ncp}}$	$\mathcal{G}_{\text{ns}}$	0.453	0.387
$\mathcal{G}_{\text{tcp}}^{\#}$	$\mathcal{G}_{\text{ts}}^{\#}$	0.519	0.963
$\mathcal{G}_{\text{tcp}}^{\#}$	$\mathcal{G}_{\text{ns}}$	0.560	0.820

Table 12.4: Numerical results for the nonlinear Predator-Prey model.

12.2.3. Revisit of parameterization of oil migration model

Although very informative, the construction and study of the contour ellipses in practice may not always be desirable or attainable, because their construction is too costly or because the function(s) underlying the model are not known. In order to test the proposed approach under these circumstances, a more complex, nonlinear, gray-box model is parameterized here. The model under study is the previously developed CA-based model for oil migration in chocolate confectionery (see Chapter 8). The model dynamics are determined by four parameters being P_t, P_d, ζ and Δt respectively (see Table 8.1) and the simulated model output is the amount of liquid fat present in different layers of the chocolate confectionery. The objective function R for this case is the SAE between the simulated model output and the experimental data at various points in time. In Chapter 8 a global SA was performed as part of the model exploration. The information gained from this SA for the four model parameters, is used to construct the FIM:

$$\text{FIM} = \begin{pmatrix} 47.45 & 22.30 & 30.81 & 44.85 \\ 22.30 & 69.16 & 48.58 & 51.68 \\ 30.81 & 48.58 & 67.31 & 54.52 \\ 44.85 & 51.68 & 54.52 & 71.90 \end{pmatrix}.$$

It is mentioned here that since the studied model is a gray-box model, the Hessian H cannot be determined, meaning that there is no way to assure that FIM is a good approximation of H for this specific case. Further, since model evaluations with the CA-based model are time consuming, the ellipsoidal contours across the search space are not checked. The eigenvalues and -vectors of FIM(R) can be computed and used to transform a naive square search grid $\mathcal{G}_{\text{ns}}$ into an improved seach grid $\mathcal{G}_{\text{ts}}^{\#}$. No CPES search grid is used here, since its construction in more than two dimensions is not trivial. It is also studied here whether the number of grid points in the search space has an influence on the performance of the approach.

Six, seven and eight coordinate values respectively, are chosen equidistantly according to each of the four dimensions in order to construct three different naive square search grids that consist out of a total of 1296 ($\mathcal{G}_{\text{ns}}^{1}$), 2401 ($\mathcal{G}_{\text{ns}}^{2}$) and 4096 ($\mathcal{G}_{\text{ns}}^{3}$) grid points, respectively. Afterwards, these grids are stretched and rotated using the information contained in the FIM as such constructing three transformed search grids that consist out of a total of 1282 ($\mathcal{G}_{\text{ts}}^{\#1}$), 2389 ($\mathcal{G}_{\text{ts}}^{\#2}$) and 4066 ($\mathcal{G}_{\text{ts}}^{\#3}$) grid points, respectively. As mentioned earlier in this chapter, it is difficult to obtain exactly the same number of search grid points for both the naive grid as for the transformed grid. Therefore, a lower total number of search grid points is chosen for the transformed search grids in order to be sure that a possible improvement in fitness stems from the proposed approach and not from the higher number of grid points in the search space.

Further, the simulations with the five best combinations of parameter values for each of the six different search grids under study are repeated ten times in order to assure that a better result obtained with a search grid is related to how the grid is constructed and not to the model stochasticity. Table 12.5 shows the performance of the different search grids expressed through two measures. A first measure is the best SAE, i.e. for this specific problem the lowest SAE, out of the five best combinations of parameter values for the different grids, averaged over the ten repetitions. The second measure is the average SAE of the five best combinations of parameter values, also averaged over the ten repetitions.

Table 12.5: Lowest average SAE from ten repetitions obtained with the five best combinations of parameter values and the average SAE obtained with the five best parameter combinations over ten repetitions.

	$\mathcal{G}_{\text{ns}}^{1}$	$\mathcal{G}_{\text{ns}}^{2}$	$\mathcal{G}_{\text{ns}}^{3}$	$\mathcal{G}_{\text{ts}}^{\#1}$	$\mathcal{G}_{\text{ts}}^{\#2}$	$\mathcal{G}_{\text{ts}}^{\#3}$
Lowest SAE	0.590	0.597	0.588	0.551	0.552	0.534
Average SAE	0.603	0.603	0.597	0.577	0.562	0.561

From Table 12.5 it is clear that the application of the proposed approach entails an advantage in terms of the SAE. For the three different search grid sizes the transformed grid, even with its lower total number of grid points, always outperforms the naive search grid in both the best found solution as well as the average SAE. The best results are obtained with $\mathcal{G}_{\mathrm{ts}}^{\#3}$, although this is mainly due to the larger number of search grid points and thus larger coverage of the search space. Nevertheless, the reduction in SAE going from $\mathcal{G}_{\mathrm{ns}}^{3}$ to is $\mathcal{G}_{\mathrm{ts}}^{\#3}$ is only marginally larger than the reduction in SAE for the smaller search grid sizes. The latter implies that for the three studied grid sizes going from a naive square search grid to a transformed search grid has approximately the same effect on the SAE, as such demonstrating the potential of the proposed approach.

12.3. Computational feasibility

Despite the potential improvement of the search grid employing the proposed approach, caution is required. A first important point that is addressed here are edge effects. For the number of grid points used throughout this chapter, edge effects do not play a significant role. Yet, when a very low number of grid points is allowed or desired, the edge effects have to be studied more carefully.

Secondly, it is pointed out that, though interesting for the visual assessment of the problem, the construction of contour plots in different parts of the search space demands a high number of model evaluations. In some cases, when doing extensive model exploration or when the same model will be employed for many different problem settings, these contour plots may be constructed despite of their cost. However, more generally, given that the goal of this paper is to get higher quality solutions for a limited number of model evaluations, it is plain that the construction of these contour plots will be computationally too costly as was the case for the CA-based model for oil migration. The latter does entail that the user cannot evaluate the shape of the contour lines and decide whether or not (s)he deems it necessary to select a subspace of the search space where contour lines are more similar. One possible way to circumvent this problem would be to study the FIM at every point in the search space that is evaluated for global SA. A judgement is then required on whether or not these FIMs deviate too much from each other in order to proceed. If they are similar, the approach described before can be adhered to. In the other case, a partitioning of the search space, based on the similarity of the FIMs, with a separate search grid for each part can be considered.

13 Conclusions

In this part of the dissertation, a method to construct a search grid when a limited number of grid points is allowed or desired, was introduced. Making use of SA information obtained during model exploration, the FIM is constructed. From this matrix, the eigenvalues and -vectors can be calculated and can be used to transform a naive search grid into a more efficient one. Through a series of computational experiments with a linear model as well as two well-known nonlinear models, it was found that when dealing with inverse problems with an objective function that gives rise to ellipsoidal contours with axes parallel to the grid axes, that a CPES grid performs better than a square grid. Furthermore, for both types of search grid there is an improvement of the results by transforming the search grid according to the proposed approach. For inverse problems with an objective function that gives rise to ellipsoidal contours with axes nonparallel to the grid axes, the best choice that can be made is either the improved CPES search grid or the naive square search grid. However, when in practice the contour plots are not constructed due to their computational cost and therefore it is not known how the contour lines are orientated with regard to the grid axes, the best choice that can be made is the improved CPES search grid. Namely, the latter grid gives the best results regardless of the orientation of the contour lines, while the naive square search grid only gives rise to the best result in one specific case. Finally, a gray-box CA-based model for oil migration was studied where the similarity of the ellipsoidal contours across the search space and the similarity of FIM and H were not investigated. Nevertheless, good results were obtained making use of the developed approach.

Obviously, the approach presented here is just a first step in the complete development of a standardized method to improve search grids. The issue of multiple parameters needs to be tackled more thoroughly. It was already suggested earlier that the approach is only useful when a limited number of parameters needs to be optimized. An extension to more parameters is necessary in order for the approach to be applicable in more complex settings. The sensitivity of the different parameters can still be used in this case to guide the parameter combinations to be evaluated, but visual assessment is hampered. Furthermore, creating a CPES search grid in higher dimensions is not trivial. Finally, modelers should investigate the computational feasability of the proposed approach for their research with special attention given to edge effects and similarity of the FIMs in the search space they consider.

PART IV

EPILOGUE

14 General conclusions and guidelines for good practice

Throughout this dissertation, various aspects of CA-based modeling for chemical engineering have been discussed. This concluding chapter will use the knowledge acquired in Parts II and III combined with the information gathered from a literature survey to draw up a set of guidelines for good practice for future modeling efforts in this field. The data for the survey were acquired by querying the Web of Science [198] for articles in English between 2000 and 2013 that dealt with CAs. From these results, a selection was made by studying the title and abstract and selecting those articles that fell into any of the following categories: chemistry, electrochemistry, polymer chemistry, food chemistry and biochemistry. A total number of 111 articles was obtained and studied. Figure 14.1 indicates that there is a growing interest in the use of CA-based models with a new maximum of publications per year in 2013. Especially the fields of reactions in aqueous environments [35, 36, 199, 200, 201, 202, 203, 204] and corrosion [159, 175, 181, 185, 205, 206, 207, 208] have received ample attention.

Nevertheless, as was already addressed in Section 3.4, there are some points of attention when using CAs. A major point is that the modeling framework for CAs is not as vast as for PDEs, for instance. To date, no general framework for designing and testing CA-based models for real-world problems and validating them using experimental data has been established [80]. This is partly due to the skepticism in the scientific community about the CA paradigm, but mainly because CA-based models are intrinsically difficult to validate. This is due to the absence of a direct link between model parameters and real-world physical variables and the intricate translation of a discrete time step to continuous time. Unfortunately, this lack of general guidelines according to which results are reported often results in incomplete reports of research efforts, making it difficult or even impossible to reproduce results. Nevertheless, in order to further spread the knowledge on CA-based models, it is crucial for researchers to present their work in a clear and understandable fashion. It is therefore the goal of this chapter not only to discuss some of the difficulties encountered when working with CA-based models and to point out some shortcomings, but moreover to provide

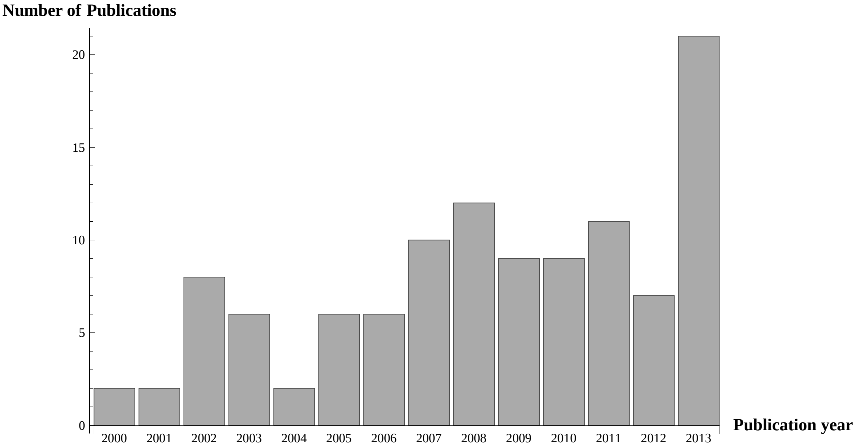


Figure 14.1: Number of publications dealing with the use of CA-based models in chemical engineering from 2000 till 2013.

some means to deal with them based on the information gathered from this dissertation. In what follows, some of the most important modeling choices when working with CA-based models are discussed and are linked to the discrete nature of a CA. Researchers should be aware of the fact that each point has to be approached rationally and should be communicated in a research paper.

14.1. Discrete space

14.1.1. Dimensionality

It is clear from Figure 14.2 that most CA-based models make use of a 2D [114, 209, 210, 211, 212, 213, 214] tessellation. Further, it can also be seen that 1D [215, 216, 217] and 3D [218, 219, 220] tessellations and combinations of tessellations of different dimensions [10, 221] discussed in the same article, exist.

A popular use of 1D CAs is the classification of proteins based on their pseudo amino acid composition [216, 217]. Here, each of the amino acids of the protein is coded in a binary way. Thereafter, this code is used to generate a 2D image function according to some CA evolving rule [26]. From these 2D figures, the geometric invariant moments are derived and

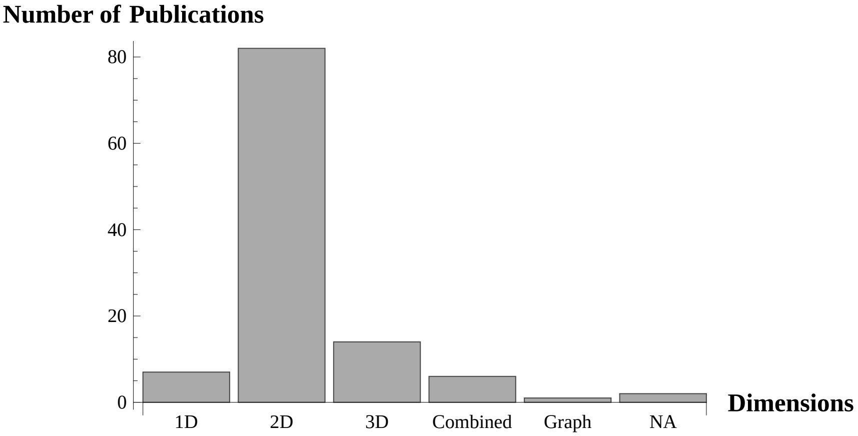


Figure 14.2: Dimension of the CA tessellation.

subsequently used in a jackknife cross-validated classification. Nevertheless, for (bio)chemical processes in general, the use of a 1D tessellation is of limited interest. On the other hand, the choice of a 3D tessellation may seem a natural one, since real-world applications also take place in a 3D space, but this choice should be made thoughtfully. The use of a 3D tessellation not only renders the implementation more difficult, it strongly extends computation times, which in practice is undesirable. Therefore, a 3D tessellation should only be adopted if the modeling objective at hand demands for higher dimensionality. For example, Yu *et al.* [218] introduce a 3D CA-based model to simulate drug release from a multi-layer biodegradable polymer microstructure, but only represent their results as a relative drug release out of the mantle in function of time. In this case, like in many other cases, a 2D tessellation suffices for the modeling purposes presented in the article. Amongst the 2D tessellations, the tessellation made up out of squares is the most popular because of its straightforward implementation. Nevertheless, in some cases a 2D tessellation made up out of hexagonal cells [206, 211] is a more advantageous option. For instance, Setny and Zacharias [200] and Mao *et al.* [222] use a hexagonal grid in order to represent molecules more naturally and distinguish between different enantiomers.

In this dissertation, mostly 2D tessellations made up out of squares were employed (see Chapters 6, 7 and 8), because they suffice for the modeling purposes at hand. Only in Chapter 9 where a CA-based modeling for pitting corrosion was developed, a 3D tessellation was used. The latter is justified since both experimental data on the affected metal surface area as the pit depth were used to parameterize the model.

14.1.2. Physical meaning of the tessellation

A second important aspect to consider regarding the space domain is that researchers have to specify what the employed tessellation actually represents. This could either be the complete physical system under study or just a (small) part of it. If the entire system is being modeled [115, 216, 223, 224], it has to be clear for the reader what each dimension of the tessellation depicts. This is particularly true when no 3D tessellation is used. For instance, Xiao *et al.* [216, 217] represent the complete amino acid sequence of a protein by means of a 1D tessellation. A natural option when a 2D tessellation is concerned, is that the tessellation represents a cross-section of the system under study [115, 223, 224]. For example, Alopaeus *et al.* [224] model liquid distribution in trickle bed reactors, where the 2D tessellation depicts a cross-section (cut parallel to the fluid flow direction) of such a reactor. Also in Chapter 8, the tessellation represented a cross-section of the confectionery model systems from which the experimental data were obtained.

Although in theory always possible, researchers should contemplate on the necessity of modeling the complete physical system. When the modeling goal does not demand it, modeling the entire system results in unnecessary calculations and excessive computation times. This holds especially true when the CA-based model deals with a well-mixed solution [225, 226, 227], such that the processes that are occurring are the same in every part of the system. In this case, a CA tessellation that only represents a part of the studied system can be used for simulations and subsequent calculations [61, 178, 228, 229]. Afterwards, the results can be translated to those for the whole system making use of a scaling factor that relates the tessellation to the real-world system. For example, Sohi and Khoshandam [225] and de Lacy Costello and Toth [226] study non-catalytic gas-solid reactions and glider guns in the light-sensitive Belousov-Zhabotinsky medium, respectively. In both cases the focus is on an elementary part of the process and not on the outcome of the experiment for the entire system under study, since the studied processes are the same everywhere. This was also illustrated in Chapters 6 and 7.

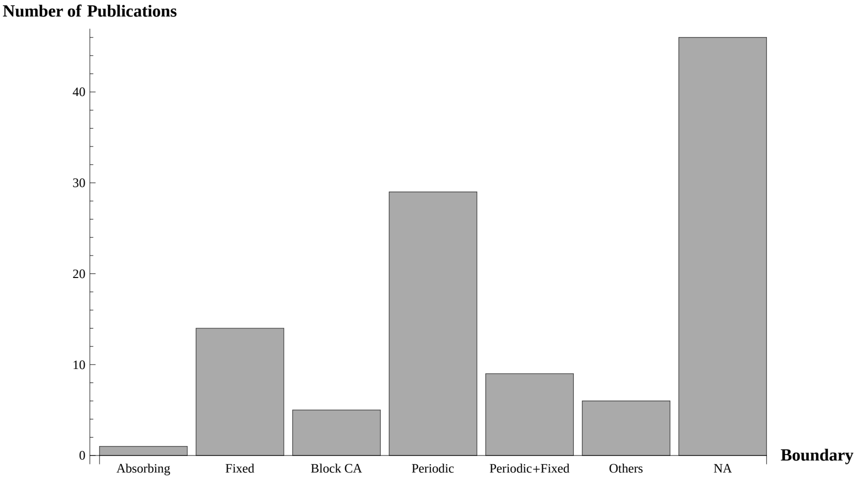


Figure 14.3: Types of boundary conditions used in studied publications.

14.1.3. Boundary conditions

Figure 14.3 shows the type(s) of boundary conditions adhered to in the studied articles. It is evident from the large portion of articles that do not mention the type of boundary condition used, that attention has to be drawn to this matter. Albeit not absolutely necessary in order to understand a presented CA-based model, a choice on the boundaries always has to be made when working with CA-based models and should therefore be reported. Furthermore, this information is crucial in order to reproduce results. Namely, the use of different boundaries, besides affecting the implementation, affects the simulated outcome.

For example, Laaksonen *et al.* [230] employed absorbing boundaries. In their CA-based model, they study drug release from a polymeric matrix. The interest of their study is in the amount of drug compounds that is released from the matrix and not what is happening with these compounds after release. Therefore, once these drug compounds reach the tessellation boundary, they are absorbed. However, in most cases, researchers are interested in the compounds (or other types of molecules) themselves and absorbing boundaries give rise to undesired loss of mass in the system. For this reason, periodic [231, 232, 233, 234] and fixed [235, 236] boundaries are more popular, as is also illustrated in Figure 14.3. The latter two boundary conditions were also adhered to in the case studies of Part II. From the point of view of implementation, the use of periodic boundaries allows the modeler to use the same neighborhood function for every cell of the tessellation, as such presenting an advantage. However, the choice between fixed and periodic boundaries is also related to what the tessellation represents (cf. supra).

In case the tessellation is a small part of a larger whole, periodic boundary conditions are frequently used, because they simulate the presence of an infinite/larger system (see Chapters 6 and 7). On the other hand, modeling the physical boundaries of a system calls for fixed boundaries since usually no movement or contact through these boundaries is possible. When a tessellation is used to model a 2D cross-section of a system, a combination of fixed and periodic boundaries [62, 65, 201] is often employed. For example, the diffusion-limited aggregation and electrodeposition of particles in the presence of leveling molecules is described by Ackland and Tweedie [62] by means of a 2D CA-based model. Deposition occurs according to the vertical axis. The top and bottom row of the 2D tessellation present fixed boundaries, since they cannot be considered as neighboring rows through periodic boundary conditions. Namely, the bottom row where deposited particles reside is qualitatively different from the top row where particles are in suspension. By contrast, the first and last column of the tessellation are qualitatively similar and therefore, periodic boundaries along the horizontal axis are employed (see Chapter 8).

14.2. Discrete state

14.2.1. Number of states

Despite the fact that the discrete nature of the state domain, together with that of the space and time domain, is at the core of CA-based modeling, Figure 14.4 demonstrates the incompleteness of a relatively large proportion of the studied articles on this subject. These articles are incomplete because either they do not mention the number of states and what the different states represent or it is unclear from the article [235, 237]. Further, some authors introduce a large number of states, making their model in practice a coupled map lattice (CML) [180, 229] (see Table 2.1) or are in fact using a CML-based model, but are incorrectly naming it a CA-based model [238, 239].

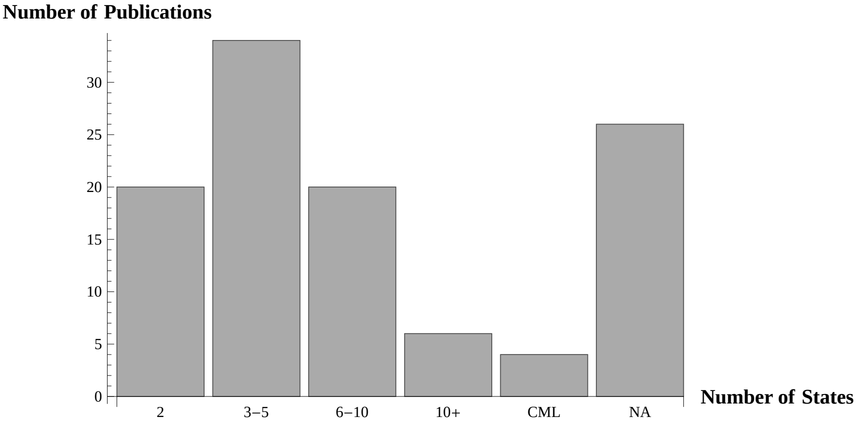


Figure 14.4: Number of different, discrete CA states in the studied publications.

Figure 14.4 also shows that a large number of researchers is limiting themselves to just two discrete states [240, 241, 242, 243]. These binary states frequently indicate the presence or absence of a compound, much like Conway's Game of Life [24]. Nevertheless, for more complex processes two states do not suffice. For that reason, the majority of researchers employs three to ten discrete states [51, 244, 245, 246]. Although it is important to assign a state to every significant component in the chemical process, the introduction of too many different states for inactive components that overcomplicate the modeling and the simulations has to be avoided.

In the different case studies of Part II the total number of discrete states was always kept below ten. In Chapters 6, 7 and 9 three discrete states were used and in Chapter 8, due to the complex composition of the confectionery model systems, seven discrete states were introduced.

14.2.2. Initial state

Like the boundary conditions, the initial state of a CA-based model has to be reported and well-argued to be able to reproduce published results. Nevertheless, some of the studied articles neglect to do so [240, 247, 248]. First of all, the number of cells in each state has to be decided on. It is logical that these numbers are derived from literature or experimental data in order to simulate the chemical process at hand more realistically. In all the discussed case studies of this dissertation, the experimental data were available which allowed for the determination of the number of cells in each of the discrete states directly from these data.

Spatially seen, an even distribution of the different discrete states is, technically, the most straightforward initial condition, which explains its popularity as initial condition in literature. However, this initial condition is not easily obtained in the laboratory [23]. In practice, this initial condition only holds true for well-mixed systems. This initial condition is also adhered to throughout this dissertation since laboratory experiments often permit to assume well-mixed systems. When not dealing with well-mixed systems, a well-considered choice has to be made on how to distribute the different discrete states over the tessellation. This can be done based on information gathered from literature on how this distribution is for the real-world phenomenon under study. Further, in some cases the discrete nature of the tessellation allows to extract the initial configuration of the tessellation directly from the experimental setup: satellite maps or tomographic images can be translated into the initial state of the CA [249, 250].

14.2.3. Model scale

A final important issue concerning the state of a cell to take into account when developing a model is the choice of the level of description, be it microscopic, mesoscopic or macroscopic. The vast majority of studied articles introduces a microscopic or mesoscopic model, with about the same number of articles describing a microscopic model as those describing a mesoscopic model. Microscopic models mostly focus on processes taking place at the atomic level, i.e. between different molecules [251, 252, 253, 254, 255]. For example, Bullard [256] proposes a model to study the early-age hydration of Ca_3SiO_5 and DeSoi *et al.* [257] developed a model to describe the chromatographic separation of compounds through HPLC. With microscopic models, the state of the cells often represents an individual molecule. Obviously, this limits the system space that can be studied without resulting in excessively large tessellation sizes and computation times.

Therefore, when the goal is to study a chemical phenomenon on a larger scale or even a complete system, while managing the computation time and computer memory use, a microscopic description is not interesting. Namely, the phenomena that are studied constitute rather scarce events on this scale. For instance, Bartosik *et al.* [178] describe a model for pitting corrosion. With the size of the corrosion pits in the range of 10-100 nm, the time and space scales of the phenomenon are far beyond the reach of the present-day atomistic level computations and, therefore, microscopic models are unfavorable for this modeling goal. On the other hand, as was mentioned in Section 3.4, a macroscopic description with classical kinetic equations, besides the fact that in most cases analytical solutions of the resulting equations are not at hand as was mentioned before, is often unable to capture the stochasticity of the underlying processes, causing that some of the information, important to engineers, is not readily available [10, 74].

The mesoscopic approach is seen as a way to combine the macroscopic phenomenology with the stochastic character of the processes originating from the microscopic scale processes [75]. CAs are suited to construct these mesoscopic models, where the goal is not to describe a specific system, but rather to analyze how a combination of a small number of basic processes, very well understood by researchers, might determine general features [236, 238, 258, 259, 260, 261, 262]. The latter entails that cells are not to be associated directly with the individual atoms or molecules, but rather with a homogeneous grouping of the same type of atom or molecule. This means that atomic size effects are not accounted for, which is a simple approximation intended to capture the synoptic effects of the process at hand [76, 77]. All developed CA-based models in this dissertation are of the mesoscopic type.

One of the studied articles, the one from Sarkar and Abbasi [263], deals with a macroscopic description of a process. Nevertheless, the subject of the article is not an actual chemical process, but rather the consequences of an accidental loss of confinement of a unit in a chemical company and the subsequent chemical hazard spread. A CA is employed here to enable one to take into account the influence of the local factors associated with each part of the tessellation since the latter in this model represents the land area surrounding the chemical company. This CA-based model is reminiscent of CA-based models for epidemic spread [264, 265].

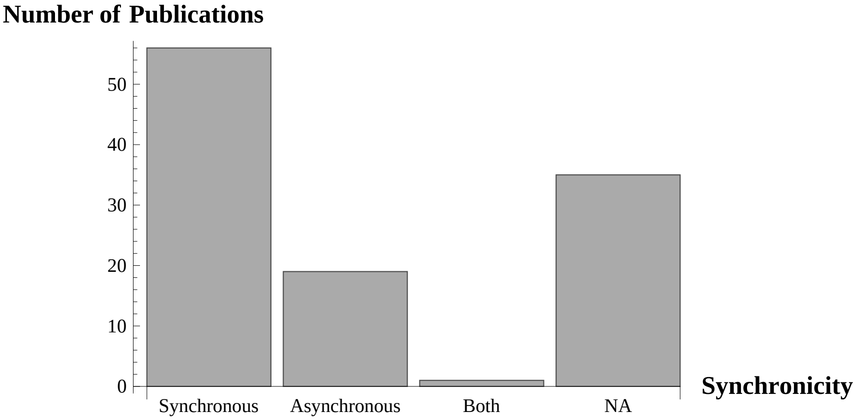


Figure 14.5: Synchronicity of updating the transition function of a CA.

14.3. Discrete time and transition function

14.3.1. Synchronicity

When performing simulations with CAs, a decision has to be made on how to apply the transition function Ψ. This can be done either synchronously or asynchronously (see Section 3.3). Figure 14.5 shows that the majority of studied articles updates the CA in a synchronous manner [266, 267, 268, 269, 270], as such adhering to the classical paradigm of CA, and a smaller portion makes use of asynchronous updating [114, 271, 272, 273]. However, more interesting is the fact that in more than 30 % of the articles no clear indication can be found in what manner the CA was updated [274, 275, 276]. Nevertheless, this information is crucial in order to reproduce results, since the choice not only has an influence on the implementation of the model but also on the simulation results.

A majority of researchers opts for the synchronous updating in their work, as it complies with the CA paradigm as conceived originally and it also entails a more straightforward implementation compared to asynchronous update methods. However, the choice for a given update method should be motivated by the process one wants to study: if a discrete time step of the CA corresponds to a relatively large time interval, expressed in physical time units, it may be assumed that every cell in $\mathcal{T}^*$ will be updated during each time step and synchronism is justified [54]. Yet, as the physical time that is covered per time step becomes smaller, the number of events that occur during its elapse decreases and asynchronism becomes a more natural option.

Throughout this dissertation, the choice for synchronous or asynchronous updating was mainly motivated by the necessity to respect the law of mass conservation on the one hand and the desire to keep calculation times under control on the other hand. In Chapter 6 a block CA was employed because it speeds up calculations, simplifies implementation, obeys mass conservation laws and eliminates the need for boundary conditions and the system under study was well-mixed. In Chapters 7 and 8 the transition function was applied in an asynchronous manner in order to respect the law of mass conservation. A block CA was not chosen to develop the models in these chapters, since the underlying phenomena were too complex for the block CA paradigm. Finally, in Chapter 9, the modified synchronous updating scheme as discussed in Section 3.3 was employed instead of an asynchronous updating scheme. The latter offers for this specific 3D model a great reduction in computation time which outweighs the issues with the limited number of update conflicts.

14.3.2. Model parameters

Only a small proportion of researchers modeling (bio)chemical processes refrains from introducing model parameters into their CA-based model [243, 255]. Despite the fact that model parameters are present in the vast majority of studied articles, the information concerning them is frequently incomplete. Figure 14.6 shows that a relative high number of researchers does not report on the exact number of parameters they employ [231, 277, 278]. Furthermore, those who do discuss their model parameters are often unclear on the exact number of parameters used or what their role and/or interpretation is [68, 270, 279]. Researchers should be highly mindful of clearly discussing all model parameters and their importance, since they are at the core of CA-based modeling of chemical processes. Normally, model parameters have a physical relevance, i.e. can be interpreted in terms of the studied process, though there are exceptions to this observation [213, 216, 217, 267].

Most researchers use one up to five parameters [254, 260, 280], which is a reasonable number of parameters to parameterize using experimental data and is also what was done in this dissertation. However, Figure 14.6 also

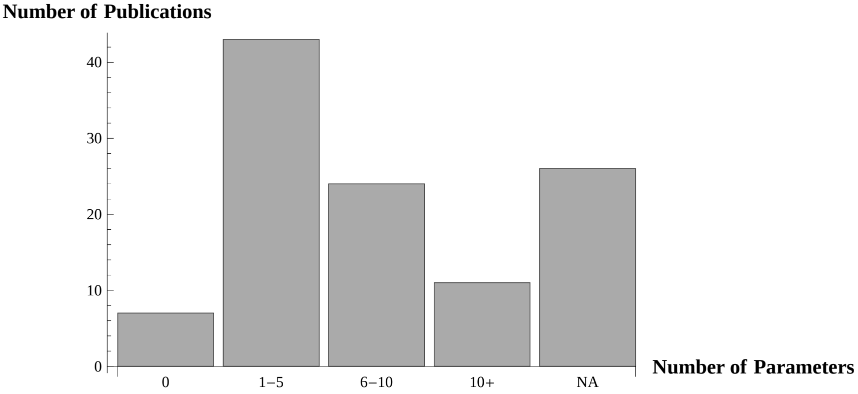


Figure 14.6: Number of model parameters in the studied publications.

shows that a substantial number of researchers develop models that contain six to ten parameters [178, 281, 282] or even more than ten parameters [211, 251, 283]. The introduction of a high number of model parameters allows to model more processes, but comes at a cost of tuning and parameterization.

Further, we want to draw attention to one specific process and its associated model parameters since it is so important in spatio-temporal modeling, namely, diffusion. In CA-based modeling, a choice has to be made on how discrete cell dimensions and a discrete time step correspond to real physical quantities. A few authors discuss the possibility to base this relationship on the diffusion coefficient [232, 246, 262], which is a measure of the speed of diffusion. When the diffusion coefficient is known, choosing a number of seconds that corresponds to a time step Δt at the same time determines the physical size of a cell Δx and vice versa (see Chapter 7). However, when no value is available for the diffusion coefficient under the circumstances that are being studied, Δx and Δt cannot be linked directly. In this case, Δx can be chosen (at random) from an acceptable range and Δt can be taken as one of the model parameters. After parameterization of the CA-based model, the optimized value of Δt can be used, together with the chosen value of Δx, to calculate an effective diffusion coefficient for that specific case (see Chapters 8 and 9).

14.3.3. Stochasticity

The last point of interest related to the transition function discussed in this chapter is the model stochasticity. About three times as many articles describe CA-based models with a stochastic transition function (see Definition 2) [284, 285, 286, 287, 288] than models with a deterministic transition function [200, 235, 289, 290]. When it comes to (bio)chemical processes, a stochastic transition function seems more logical since it captures the randomness of these processes in a natural manner. However, the latter implies that simulations starting with the same parameter values and initial condition, can result in different simulation results. This entails the need for repeated simulations to be able to interpret the results correctly. All CA-based models of Part II are stochastic models.

The stochasticity of the transition function is often incorporated in the model parameters that therefore usually can be interpreted as a measure of the reaction constants of the underlying (bio)chemical process [10, 115]. For instance, in the CA-based corrosion model described by Córdoba-Torres *et al.* [285], the transition function is stochastic and the model parameters are derived from kinetic parameters of the different studied corrosion mechanisms. The optimized parameter values are thereafter used to make predictions on the spatial statistical patterns on the metal surface. Besides the interpretation as reaction constants, model parameters can further be employed to manage the difference in time scale between different processes. By attributing to process A a probability of occurrence that is only a third of the probability of occurrence of process B, process A effectively proceeds at a third of the speed of process B.

As seen from the different case studies of Part II, the stochasticity of the developed models entails the necessity for repetition of the model evaluations to account for the variation in the simulated results. Given the fact that the model evaluations in this dissertation roughly range from 5 min up to 1 day, depending on the employed combination of parameter values (especially the value for Δt is of importance), attention should be spent on the choice of the number of repetitions to perform. Selecting a high number of repetitions, e.g. 100 repetitions, is a safe choice and is feasible when only the simulated output for a limited number of combinations of parameter values is required. Nevertheless, for the parameterisation performed in the different case studies of Part II a high number of combinations of parameter

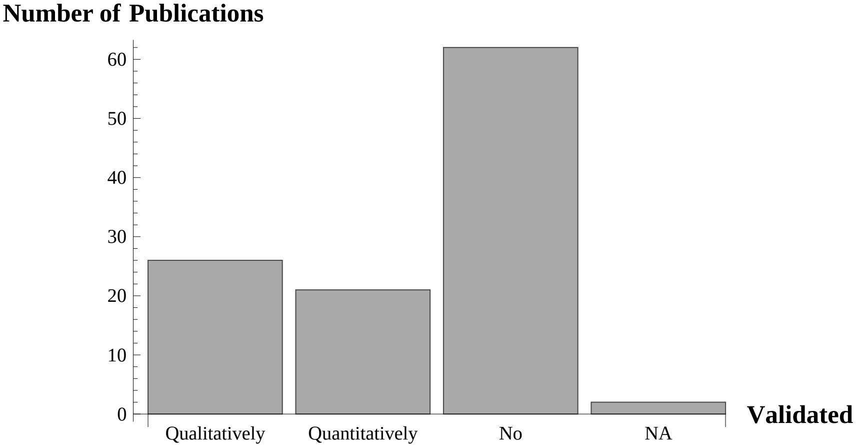


Figure 14.7: Model validation of the studied publications.

values is tested. In this case, 100 repetitions of each combination would result in excessive calculation times. Therefore, a low number of repetitions, e.g. between five and ten, should be chosen and simulated results analyzed. Only when too much variation is observed between the results stemming from the low number of repetitions, should the number of repetitions be raised.

14.4. Validation

From the literature survey, it is clear that for a modeling paper in general and one making use of a CA-based model specifically to get published, calibration and/or validation of the developed model is strictly seen not essential. However, this section stresses the importance of a sound validation of the model. Validating a model increases the credibility of the developed model as well as gives an idea about the practicality of the model. Figure 14.7 shows that more than half of the studied articles forgo model validation of any sort [61, 208, 248, 287]. This implies that there is no way of assuring that the simulated results with the CA-based model bear any resemblance to the real-world phenomenon.

Among the validated models, the majority of researchers discusses a qualitative validation, i.e. a visual assertion of the resemblance of the simulations and the studied process [69, 222, 267, 268]. For example, Zhao *et* *al.* [267] study the dendritic growth of NH_4Br crystals with a CA-based model. They pre-process real images taken with a CCD camera and visually compare these to simulated results to prove the validity of their binary CA-based model. However, more interesting is the validation of a model using quantitative experimental data [10, 115, 213, 256]. For instance, Cao *et al.* [213] developed a CA-based model to investigate the diffusion of chloride ions in concrete. They had access to time series of chloride concentrations in concrete and compared these results with their simulations as well as with simulations obtained by making use of Fick's second law of diffusion. They saw an improvement by using their proposed model over the classical approach with Fick's law. One step further is the use of experimental data to optimize the values of the model parameters such that the simulated results are as close as possible to the experimental data. For instance, Lishchuk *et al.* [10] predict intergranular corrosion by means of a 3D CA-based model. They parameterize their model making use of experimental data on intergranular corrosion. The employed optimization algorithm in this case was based on a combination of genetic algorithms and the Nelder and Mead method.

All models in Part II were parameterized using experimental data, which allowed for both a check of the model's correctness and an interpretation of the optimized parameters. In Part III of this dissertation, a method to construct a search grid when a limited number of grid points is allowed or desired making use of SA information, was introduced. The latter is of great use when CA-based models for chemical engineering problems are concerned.

14.5. Checklist for CA-based modeling

As was illustrated from the various examples, too many papers are lacking in one or more of the critical points of information addressed in this chapter, rendering them difficult or even impossible to implement just based on the information present in the paper. For that reason, this chapter concludes with a summary of the different critical points to take into consideration in the form of a checklist that can be used to guide future modelers. Figure 14.8 illustrates the different points discussed in this chapter and the most important considerations to take into account. Researchers should be

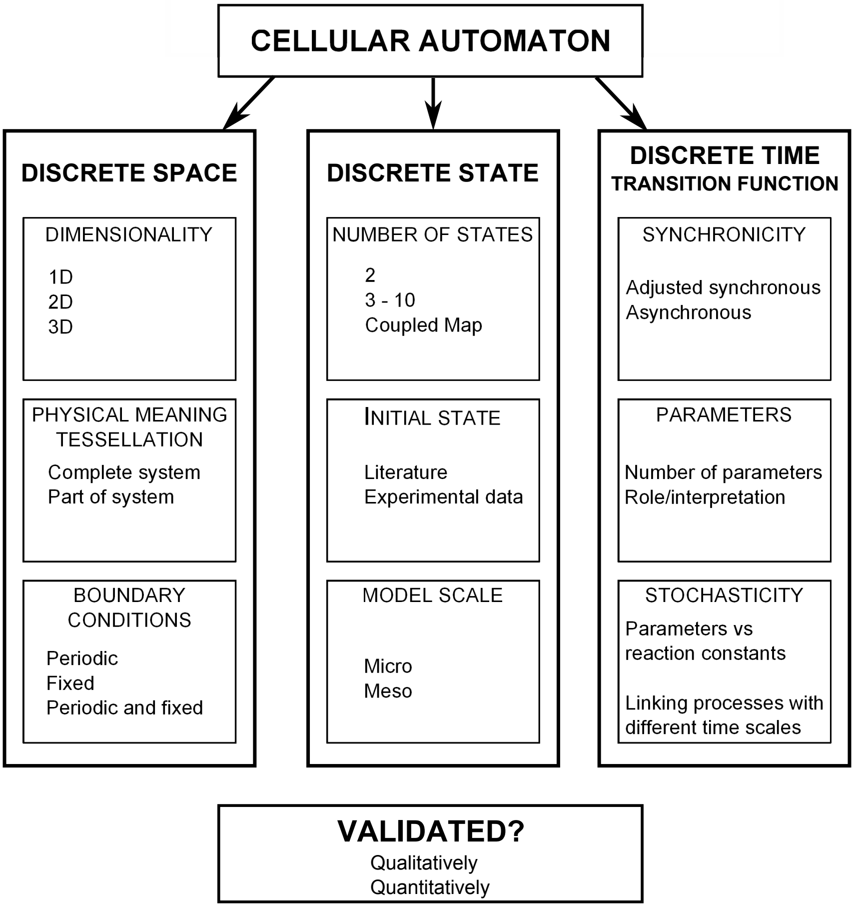


Figure 14.8: Checklist with necessary points of information for CA modeling.

aware of each of these critical points, make thought-out choices concerning them and report these choices in their paper. This will hopefully improve the quality of research papers on CA-based modeling and help spread the knowledge on these interesting models.

Dutch summary

Wiskundige modellen zijn een vertaling van een natuurlijk of artificieel systeem naar een stel wiskundige vergelijkingen. Het gebruik van modellen biedt een aantal voordelen zoals een tijdswinst ten opzichte van laboratoriumexperimenten en de mogelijkheid tot het testen van hypotheses. Verschillende problemen vereisen verschillende aanpakken en daardoor ook verschillende modellen. In het domein van de (bio)chemie spelen voornamelijk spatio-temporele processen, en daardoor ook spatio-temporele modellen, een belangrijke rol. De meest gebruikte en ook klassieke manier voor het modelleren van dergelijke processen is aan de hand van (partiële) differentiaalvergelijkingen, die continu zijn in hun ruimte-, tijds- en toestandsdomein, wat geleid heeft tot een diepgaande kennis over dit soort vergelijkingen, alsook tot de beschikbaarheid van een uitgebreide set aan tools voor het bestuderen van modellen die gebaseerd zijn op differentiaalvergelijkingen. Echter, (partiële) differentiaalvergelijkingen zijn niet het enige beschikbare paradigma voor het modelleren van (bio)chemische processen. Zo bestaan er o.a. coupled map lattices, agent-gebaseerde modellen en cellulaire automaten. Deze andere paradigma's worden de laatste jaren steeds meer bestudeerd om een aantal problemen of tekortkomingen van modellen gebaseerd op differentiaalvergelijkingen te overwinnen. Eén van deze paradigma's, dat tevens het onderwerp vormt van deze dissertatie, is de cellulaire automaat (CA), die discreet is in ruimte, tijd en toestand.

Een CA is een verzameling van cellen die een tessellatie vormt waarbij elke cel haar toestand aanpast op basis van een gegeven transitiefunctie die gebruik maakt van de huidige toestand van de cel en die van zijn buren. Deze transitiefunctie wordt iteratief toegepast voor een vooraf bepaald aantal tijdstappen. Ondanks de groeiende interesse in CA-gebaseerde modellen is er nog steeds maar een beperkt aantal publicaties in het domein van de (bio)chemie waarin dit type modellen gehanteerd wordt. Bovendien is de rapportering ervan, wanneer deze modellen gebruikt worden, vaak onvolledig. Dit is grotendeels te wijten aan het ontbreken van een algemeen kader volgens hetwelk dergelijke modellen opgebouwd en gebruikt kunnen worden. Het is daarom het doel van deze thesis om een eerste stap te zetten in het ontwikkelen van een dergelijk kader.

In Deel I van deze thesis wordt een formele definitie aangeleverd van een CA, wordt dieper ingegaan op het onderscheid tussen synchroon en asynchroon toepassen van de transitiefunctie en worden de voor- en nadelen van CA's besproken. Verder wordt in dit deel de keuze voor de Elementaire Elementen methode, een variantiegebaseerde scanningmethode, als sensitiviteitsanalysetechniek toegelicht voor de gevalstudies in deze thesis.

Deel II van de dissertatie probeert de voordelen van CA-gebaseerd modelleren te benutten en de beperkingen ervan te overwinnen. Vier gevalstudies worden in dit deel uitgebreid behandeld. Elk van de gevalstudies is verschillend zowel in het proces dat gemodelleerd wordt als in de aspecten van CA-gebaseerde modellering die onderzocht worden. Op deze manier wordt zowel een overzicht gegegeven van de mogelijke toepassingsdomeinen, alsook een meer gedetailleerde beschrijving van welke modelleerkeuzes kunnen gemaakt worden om bepaalde praktische limitaties van CA-gebaseerde modellen aan te pakken. De vier gevalstudies die besproken worden zijn reactiekinetiek, fotokatalyse, oliemigratie in chocolade en putcorrosie in metaal.

Elk van de hoofdstukken in Deel II is op een gelijkaardige manier gestructureerd. In de eerste plaats worden het chemisch process dat het onderwerp van dat hoofdstuk vormt, alsook de eerdere modellen in dit veld verder uitgediept. Vervolgens wordt de datacollectie besproken, waarna een groot deel van het hoofdstuk gewijd wordt aan de ontwikkeling van het model en de modelleerkeuzes die daarbij gemaakt moeten worden. Alle modellen ontwikkeld in deze dissertatie zijn stochastische modellen, hetgeen inhoudt dat de transitiefunctie gestuurd wordt door een aantal modelparameters die verbonden zijn met reactieprobabiliteiten eigen aan het bestudeerde proces. Daarom zal, eens de modellen ontwikkeld zijn, de vergaarde experimentele data gebruikt worden om de parameterwaarden te bepalen die aanleiding geven tot modelsimulaties die zich dicht mogelijk aanleunen bij de experimentele data, ook parameterisatie genoemd. Elk hoofdstuk sluit tenslotte af met een aantal conclusies en suggesties van verder onderzoek waarbij aandacht besteed wordt aan de modelleerkeuzes die gemaakt werden in dat hoofdstuk.

Uit de verschillende gevalstudies van Deel II is duidelijk dat een efficiënte parameterisatie van de verschillende CA-gebaseerde modellen geen eenvoudige taak is. Daarom wordt in Deel III een verbeterd zoekgrid voor parameterisatie voorgesteld gebruikmakend van de resultaten van een sensitiviteitsanalyse. Eerst wordt de theorie achter deze aanpak in detail besproken, waarna een aantal computationele experimenten met een eenvoudig lineair model en twee welgekende niet-lineaire modellen uitgevoerd worden om de validiteit van de voorgestelde aanpak te onderzoeken. Met behulp van de resultaten van een sensitiviteitsanalyse wordt de Fisher informatiematrix opgesteld waarvan de eigenwaarden en -vectoren gebruikt worden om een naïef zoekgrid om te vormen tot een meer doordacht zoekgrid. Het blijkt dat wanneer het een parameterisatieprobleem betreft waarvan de doelfunctie aanleiding geeft tot ellipsoïdale contouren met assen parallel aan de zoekgridassen, dat een bolstapelrooster beter presteert dan een vierkant zoekgrid. Bovendien kunnen voor beide zoekgridtypes betere resultaten bekomen worden door ze om te vormen via de voorgestelde aanpak. Voor parameterisatieproblemen met doelfuncties die aanleiding geven tot ellipsoïdale contouren die niet parallel zijn aan de zoekgridassen, is opnieuw het getransformeerde bolstapelrooster de beste keuze al benaderen de resultaten van het naïef vierkant zoekgrid deze optimale resultaten. Tenslotte wordt ook de parameterisatie van het CA-gebaseerd model voor oliemigratie uit Deel II herhaald waarbij tevens goede resultaten bekomen worden.

In het vierde en laatste deel van deze dissertatie worden een aantal conclusies meegegeven. Deze laatste zijn geschreven in de vorm van een checklist (zie Figuur 14.9) die kan gevolgd worden wanneer men een CA-gebaseerd model voor (bio)chemische processen wil ontwikkelen en is opgesteld door een combinatie van de kennis opgedaan uit Delen II en III van deze dissertatie en een literatuuronderzoek omtrent bestaande CA-gebaseerde modellen voor dergelijke toepassingen.

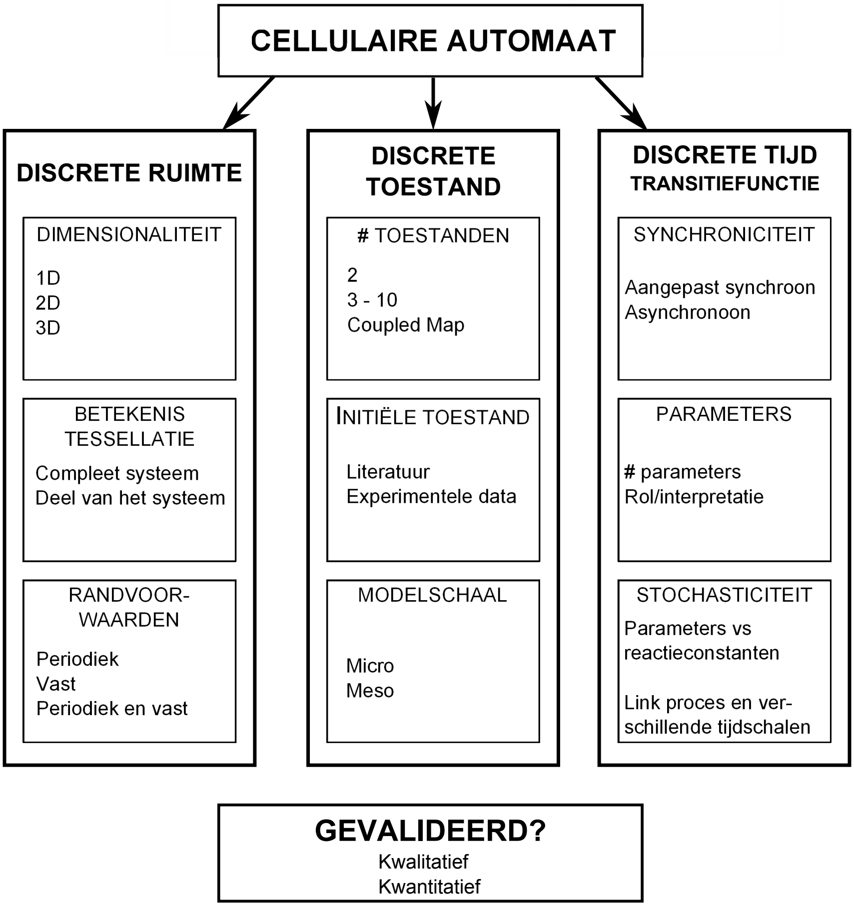


Figure 14.9: Checklist met aandachtpunten voor CA-gebaseerde modellering van (bio)chemische processen.

Bibliography

[1] A. V. Antoniouk, R. V. N. Melnik (Eds.), Mathematics and life sciences, De Gruyter, 2013.

[2] L. Michaelis, M. L. Menten, Die Kinetik der Invertinwirkung, Biochem. Z 49 (1913) 333–369.

[3] J. B. S. Haldane, The combination of linkage values, and the calculation of distances between the loci of linked factors, J. Genet. 8 (1919) 299–309.

[4] J. D. Murray, Mathematical Biology I: An Introduction, Springer-Verlag, 2003.

[5] A. L. Hodgkin, A. F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve, J. Physiol. 117 (1952) 500–544.

[6] R. Aris, Mathematical Modelling Techniques, Dover Publications Inc., 1994.

[7] E. A. Bender, An Introduction to Mathematical Modeling, Dover Publications Inc., 2000.

[8] N. Gershenfeld, The Nature of Mathematical Modeling, Cambridge University Press, 1998.

[9] J. L. Shearer, B. T. Kulakowski, J. F. Gardner, Dynamic Modeling and Control of Engineering Systems, Prentice-Hall, Inc., 1997.

[10] S. V. Lishchuk, R. Akid, K. Worden, J. Michalski, A cellular automaton model for predicting intergranular corrosion, Corros. Sci. 53 (2011) 2518–2526.

[11] R. K. Standish, Why Occam's razor, Found. Phys. Lett. 17 (2004) 255–266.

[12] N. A. C. Cressie, Statistics for Spatial Data, John Wiley, New York, 1993.

[13] N. A. C. Cressie, C. K. Wikle, Statistics for Spatio-Temporal Data, Wiley, New York, 2011.

[14] P. J. Brockwell, R. A. Davis, Introduction to Time Series and Forecasting, Springer-Verlag, 2002.

[15] L. Kier, P. Seybold, C. Cheng, Modelling Chemical Systems using Cellular Automata, Springer, Dordrecht, The Netherlands, 2005.

[16] M. P. Allen, D. J. Tildesley, Computer Simulation in Chemical Physics, Kluwer Academic Publishers, 1998.

[17] A. R. Leach, Molecular Modelling: Principles and Application, Longman, Harlow, 1996.

[18] S. Feiz, A. H. Navarchian, Emulsion polymerization of styrene: Simulation the effects of mixed ionic and non-ionic surfactant system in the presence of coagulation, Chem. Eng. Sci. 69 (2012) 431–39.

[19] F. Logist, J. Van Impe, Multi-objective dynamic optimisation of cyclic chemical reactors with distributed parameters, Chem. Eng. Sci. 80 (2012) 429–434.

[20] J. R. Weimar, J. J. Tyson, L. T. Watson, Diffusion and wave propagation in cellular automata models for excitable media, Physica D 55 (1992) 309–327.

[21] S. El Yacoubi, A. El Jai, Cellular automata and spreadability, Math. Comput. Modell. 36 (2002) 1059–1074.

[22] T. Toffoli, Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics, Physica D 10 (1984) 117–127.

[23] L. Berec, Techniques of spatially explicit individual-based models: construction, simulation and mean-field analysis, Ecol. Modell. 150 (2002) 55–81.

[24] M. Gardner, Mathematical Games - The fantastic combinations of John Conway's new solitaire game 'life', Sci. Am. 223 (1970) 120–123.

[25] S. Wolfram, Statistical mechanics of cellular automata, Rev. Mod. Phys. 55 (1983) 601–644.

[26] S. Wolfram, A New Kind of Science, Wolfram Media Inc., Champaign, United States, 2002.

[27] S. Darbha, K. Rajagopal, V. Tyagi, A review of mathematical models for the flow of traffic and some recent results, Nonlinear Anal-Theor. 69 (2008) 950–970.

[28] Y. Fan, S. Ying, B. Wang, Y. Wei, The effect of investor psychology on the complexity of stock market: an analysis based on cellular automaton model, Comput. Ind. Eng. 56 (2009) 69–79.

[29] S. Alves, N. Neto, M. Martins, Electoral survey's influence on the voting processes: a cellular automata model, Physica A 316 (2002) 601–614.

[30] S. Succi, The Lattice Boltzmann Equation, for Fluid Dynamics and Beyond, Oxford Science Publications, 2001.

[31] D. A. Wolf-Gladrow, Lattice-Gas Cellular Automata and Lattice Boltzmann Models, Springer, 2000.

[32] B. Chopard, M. Droz, Cellular automata model for the diffusion equation, J. Stat. Phys. 64 (1991) 859–892.

[33] B. Chopard, M. Droz, Cellular Automata Modeling of Physical Systems, University Press, Cambridge, United Kingdom, 1998.

[34] L. Kier, C. Cheng, A cellular automata model of diffusion in aqueous systems, Chem. Biodivers. 4 (2007) 2555–2563.

[35] L. Kier, A cellular-automata model of the structure of bulk water, Chem. Biodivers. 4 (2007) 2540–2546.

[36] L. Kier, C. Cheng, Models of solute aggregation using cellular automata, Chem. Biodivers. 6 (2009) 396–401.

[37] R. Field, F. Schneider, Oscillating chemical reactions and nonlinear dynamics, J. Chem. Educ. 66 (1989) 195–204.

[38] J. Schiff, Cellular Automata: A Discrete View of the World, John Wiley & Sons Ltd., Chichester, United Kingdom, 2008.

[39] J. Greenberg, B. Hassard, S. Hastings, Pattern formation and periodic structures in systems modeled by reaction-diffusion equations, B. Am. Math. Soc. 84 (1978) 1296–1327.

[40] Y. Zhao, S. Billings, A. Routh, Identification of the Belousov-Zhabotinskii reaction using cellular automata, Int. J. Bifurcat. Chaos 17 (2007) 6605–6615.

[41] V. Ghosh, G. R. Ziegler, R. C. Anantheswaran, Fat, moisture, and ethanol migration through chocolates and confectionary coatings, Crit. Rev. Food Sci. Nutr. 42 (2002) 583–626.

[42] S. V. Krivovichev, Actinyl compounds with hexavalent elements (S, Cr, Se, Mo) - Structural diversity, nanoscale chemistry, and cellular automata modeling, Eur. J. Inor. Chem. 18 (2010) 2594–2603.

[43] X. Xiao, S. Shao, Y. Ding, Z. Huang, X. Chen, K. C. Chou, Using cellular automata to generate picture representation for biological sequences, Amino Acids 28 (2005) 29–35.

[44] E. Reis, L. Santos, S. Pinho, A cellular automata model for avascular solid tumor growth under the effect of therapy, Physica A 388 (2009) 1303–1314.

[45] P. Gerlee, A. Anderson, A hybrid cellular automaton model of clonal evolution in cancer: The emergence of the glycolytic phenotype, J. Theor. Biol. 250 (2008) 705–722.

[46] P. Siregar, M. Chahine, F. Lemoulec, P. Lebeux, An interactive qualitative model in cardiology, Comput. Biomed. Res. 28 (1995) 443–478.

[47] G. Pizarro, D. Griffeath, D. Noguera, Quantitative cellular automaton model for biofilms, J. Environ. Eng. 127 (2001) 782–789.

[48] C. Picioreanu, M. van Loosdrecht, J. Heijnen, Mathematical modelling of biofilm structure with a hybrid differential-discrete cellular automaton approach, Biotechnol. Bioeng. 58 (1998) 101–116.

[49] R. C. Martins, V. V. Lopes, Modelling supercooling in frozen strawberries: Experimental analysis, cellular automation and inverse problem methodology, J. Food Eng. 80 (2007) 126–141.

[50] D. Mallet, L. De Pillis, A cellular automata model of tumor-immune system interactions, J. Theor. Biol. 239 (2006) 334–350.

[51] K. Jahns, M. Landwehr, J. Wübbelmann, U. Krupp, Numerical analysis of internal oxidation and nitridation by the cellular automata approach, Ocid. Met. 79 (2013) 107–120.

[52] C. S. Laspidou, A. Kungolos, P. Samaras, Cellular-automata and individual-based approaches for the modeling of biofilm structures: Pros and cons, Desalination 250 (2010) 390–394.

[53] J. M. Baetens, B. De Baets, Phenomenological study of irregular cellular automata based on lyapunov exponents and jacobians, Chaos 20 (2010) 033112.

[54] B. Schönfisch, A. de Roos, Synchronous and asynchronous updating in cellular automata, Biosystems 51 (1999) 123–143.

[55] G. D. Ruxton, L. A. Saravia, The need for biological realism in updating of cellular automata models, Ecol. Modell. 107 (1998) 105–112.

[56] L. B. Kier, C. Cheng, B. Testa, Cellular automata models of biochemical phenomena, Future Gener. Comp. Sys. 16 (1999) 273–289.

[57] S. Adachi, F. Peper, J. Lee, Computation by asynchronously updating cellular automata, J. Stat. Phys. 114 (2004) 261–289.

[58] J. Lee, S. Adachi, F. Peper, K. Morita, Asynchronous Game of Life, Physica D 194 (2004) 369–384.

[59] J. O. Indekeu, C. V. Giuraniuc, Cellular automaton for bacterial towers, Physica A 336 (2004) 14–26.

[60] S. Bandini, A. Bonomi, G. Vizzari, What do we mean by asynchronous CA? A reflection on types and effects of asynchronicity, Lect. Notes Comput. Sci. 6350 (2010) 385–394.

[61] F. Haack, K. Burrage, R. Redmer, A. M. Uhrmacher, Studying the role of lipid rafts on protein receptor bindings with cellular automata, IEEE ACM T. Comput. Bi. 10 (2013) 760–770.

[62] G. J. Ackland, E. S. Tweedie, Microscopic model of diffusion limited aggregation and electrodeposition in the presence of leveling molecules, Phys. Rev. E 73 (2006) 011606.

[63] T. Toffoli, N. Margolus, “II.12 The Margolus neighborhood”, Cellular Automata Machines: A New Environment for Modeling, MIT Press, 1987.

[64] X.-S. Yang, Turing pattern formation of catalytic reaction-diffusion systems in engineering applications, Modelling Simul. Mater. Sci. Eng. 11 (2003) 321–329.

[65] I. L'Heureux, S. Katsev, Oscillatory zoning in a (Ba,Sr)SO4 solid solution: Macroscopic and cellular automata models, Chem. Geol. 225 (2006) 230–243.

[66] W. Wang, J. L. Murray, S. Y. Hu, L. Q. Chen, H. Weiland, Modeling of plate-like precipitates in aluminum alloys-comparison between phase field and cellular automaton methods, J. Phase Equilib. Diff. 28 (2007) 285–264.

[67] T.-H. Kim, S. H. Jung, K.-H. Cho, Investigations into the design principles in the chemotactic behavior of *Escherichia coli*, Biosystems 91 (2008) 171–182.

[68] B. Pathi, S. T. Kinsey, B. R. Locke, Influence of reaction and diffusion on spatial organization of mitochondria and effectiveness factors in skeletal muscle cell design, Biotechnol. Bioeng. 108 (2011) 1912–1924.

[69] H. Wang, E.-H. Han, Simulation of metastable corrosion pit development under mechanical stress, Electrochem. Acta. 90 (2013) 128–134.

[70] D. Siegismund, A. Schroeter, S. Schuster, M. Rettenmayr, Quantitative modeling of fibrinogen adsorption on different biomaterials, Cell. Mol. Bioeng. 6 (2013) 210–219.

[71] A. Ilachinski, Cellular Automata, A Discrete Universe, World Scientific, London, United Kingdom, 2001.

[72] O. Bandman, Theory and Applications of Cellular Automata, Luniver Press, United Kingdom, 2008, Ch. Mapping physical phenomena onto CA-models, pp. 381–397.

[73] O. Bandman, Cellular-neural automaton: a hybrid model for reaction-diffusion simulation, Future Gener. Comp. Sy. 18 (2002) 737–745.

[74] D. di Caprio, C. Vautrin-Ul, J. Stafiej, J. Saunier, A. Chaussé, D. Féron, J. P. Badiali, Morphology of corroded surfaces: Contribution of cellular automaton modelling, Corros. Sci. 53 (2011) 418–425.

[75] A. Salcido (Ed.), Cellular Automata - Innovative Modelling for Science and Engineering, InTech, 2011.

[76] C. G. Chase, Fluvial landsculpting and the fractal dimension of topography, Geomorphology 5 (1992) 39–57.

[77] C. Vautrin-Ul, H. Mendy, A. Taleb, A. Chaussé, J. Stafiej, J. P. Badiali, Numerical simulations of spatial heterogeneity formation in metal corrosion, Corros. Sci. 50 (2008) 2149–2158.

[78] M. Kuntz, P. Lavallee, J. C. Mareschal, Determination of elastic properties of very heterogeneous media with cellular automata, J. Geophys. Res-Sol. Ea. 102 (1997) 7647–7658.

[79] S. G. Berjak, J. W. Hearne, An improved cellular automaton model for simulating fire in a spatially heterogeneous savanna system, Ecol. Modell. 148 (2002) 133–151.

[80] V. Grimm, E. Revilla, U. Berger, F. Jeltsch, W. M. Mooij, S. F. Railsback, H. Thulke, J. Weiner, T. Wiegand, D. L. DeAngelis, Pattern-oriented modeling of agent-based complex systems: Lessons from ecology, Science 310 (2005) 987–991.

[81] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, Global Sensitivity Analysis: The Primer, Wiley-Interscience, 2008.

[82] A. Mokhtari, H. C. Frey, Sensitivity analysis of a two-dimensional probabilistic risk assessment model using analysis of variance, Risk Anal. 25 (2005) 1511–1529.

[83] A. Saltelli, M. Ratto, S. Tarantola, F. Campolongo, Sensitivity analysis for chemical models, Chem. Rev. 105 (2005) 2811–2827.

[84] P. M. Schlosser, Experimental design for parameter estimation through sensitivity analysis, J. Toxicol. Environ. Health 43 (1994) 495–530.

[85] A. B. Massada, Y. Carmel, Incorporating output variance in local sensitivity analysis for stochastic models, Ecol. Modell. 213 (2008) 463–467.

[86] D. J. W. De Pauw, P. A. Vanrolleghem, Practical aspects of sensitivity function approximation for dynamic models, Math. Comput. Modell. 12 (2006) 395–414.

[87] A. van Griensven, T. Meixner, S. Grunwald, T. Bishop, M. Diluzio, R. Srinivasan, A global sensitivity analysis tool for the parameters of multi-variable catchment models, J. Hydrol. 324 (2006) 10–23.

[88] M. D. Morris, Factorial sampling plans for preliminary computational experiments, Technometrics 33 (1991) 161–174.

[89] H. Liu, F. Yang, C. Huang, H. Fang, Y. Cheng, Sensitivity analysis of the semi-empirical model for the growth of the indigenous *Acidithiobacillus thiooxidans*, Chem. Eng. J. 129 (2007) 105–112.

[90] M. V. Ruano, J. Ribes, J. Ferrer, G. Sin, Application of the Morris method for screening the influential parameters of fuzzy controllers applied to wastewater treatment plants, Water Sci. Technol. 63 (2011) 2199–2206.

[91] F. Campolongo, J. Cariboni, A. Saltelli, An effective screening design for sensitivity analysis of large models, Environ. Modell. Softw. 22 (2007) 1509–1518.

[92] V. Kocabas, S. Dragicevic, Assesing cellular automata model behaviour using a sensitivity analysis approach, Comput. Environ. Urban 30 (2006) 921–953.

[93] N. Garijo, R. Manzano, R. Osta, M. A. Perez, Stochastic cellular automata model of cell migration, proliferation and differentiation: Validation with *in vitro* cultures of muscle satellite cells, J. Theor. Biol. 314 (2012) 1–9.

[94] I. Merelli, D. Pescini, E. Mosca, P. Cazzaniga, C. Maj, G. Mauri, L. Milanesi, Grid computing for sensitivity analysis of stochastic biological models, Lect. Notes Comput. Sc. 6873 (2011) 62–73.

[95] P. J. Brockwell, R. A. Davis, Time Series: Theory and Methods, Springer Series in Statistics, 1986.

[96] C. M. Guldberg, P. Waage, Studies concerning affinity, J. Chem. Educ. 63 (1986) 1044–1047.

[97] T. Reichenbach, M. Mobilia, E. J. Frey, Self-organization of mobile populations in cyclic competition, Theor. Biol. 254 (2008) 368–383.

[98] C. V. Nicholas, M. A. Desai, P. M. Vadgama, Measurement of effective HCl diffusion-coefficients through aqueous and gel films by a pH jump technique, J. Chem. Soc. Faraday Trans. 87 (1991) 293–296.

[99] T. Paul, M. C. Dodd, T. J. Strathmann, Photolytic and photocatalytic decomposition of aqueous ciprofloxacin: Transformation products and residual antibacterial activity, Water Res. 44 (2010) 3121–3132.

[100] E. Rubinstein, History of quinolones and their side effects, Chemotherapy 47 (2001) 3–8.

[101] A. Fujishima, T. N. Rao, D. A. Tryk, Titanium dioxide photocatalysis, J. Photochem. Photobiol. C 1 (2000) 1–21.

[102] U. I. Gaya, A. H. Abdullah, Heterogeneous photocatalytic degradation of organic contaminants over titanium dioxide: A review of fundamentals, progress and problems, J. Photochem. Photobiol. C 9 (2008) 1–12.

[103] O. Carp, C. L. Huisman, A. Reller, Photoinduced reactivity of titanium dioxide, Prog. Solid State Chem. 32 (2004) 33–177.

[104] U. Diebold, The surface science of titanium dioxide, Surf. Sci. Rep. 48 (2003) 53–229.

[105] X. Van Doorslaer, K. Demeestere, P. M. Heynderickx, H. Van Langenhove, J. Dewulf, UV-A and UV-C induced photolytic and photocatalytic degradation of aqueous ciprofloxacin and moxifloxacin: Reaction kinetics and role of adsorption, Appl. Catal. B. 101 (2011) 540–547.

[106] X. Van Doorslaer, P. M. Heynderickx, K. Demeestere, K. Debevere, H. Van Langenhove, J. Dewulf, TiO_2 mediated heterogeneous photocatalytic degradation of moxifloxacin: Operational variables and scavenger study, Appl. Catal. B. 112 (2012) 150–156.

[107] A. J. Maira, W. N. Lau, C. Y. Lee, P. L. Yue, C. K. Chan, K. L. Yeung, Performance of a membrane-catalyst for photocatalytic oxidation of volatile organic compounds, Chem. Eng. Sci. 58 (2003) 959–962.

[108] S. Cao, K. L. Yeung, P. Yue, An investigation of trichloroethylene photocatalytic oxidation on mesoporous titania-silica aerogel catalysts, Appl. Catal., B. 76 (2007) 64–72.

[109] U. Diebold, N. Ruzycki, G. S. Herman, A. Selloni, One step towards bridging the materials gap: surface studies of TiO_2 anatase, Catalysis Today 85 (2003) 93–100.

[110] A. L. Linsebigler, G. Lu, J. T. Yates, Photocatalysis on TiO_2 surfaces: Principles, mechanisms and selected results, Chem. Rev. 95 (1995) 735–758.

[111] A. Maudhuit, C. Raillard, V. Hequet, L. Le Coq, J. Sablayrolles, L. Molins, Adsorption phenomena in photocatalytic reactions: The case of toluene, acetone and heptane, Chem. Eng. J. 170 (2011) 464–470.

[112] R. Inam, H. Mercan, E. Yilmaz, B. Uslu, Differential pulse polarographic determination of moxifloxacin hydrochloride in pharmaceuticals and biological fluids, Analytical Letters 40 (2007) 529–546.

[113] S. Malato, P. Fernandez-Ibanez, M. I. Maldonado, J. Blanco, W. Gernjak, Decontamination and disinfection of water by solar photocatalysis: Recent overview and trends, Catalysis Today 147 (2009) 1–59.

[114] P. Van der Weeën, J. M. Baetens, J. Verwaeren, X. Van Doorslaer, P. M. Heynderickx, J. Dewulf, B. De Baets, Modeling the photocatalytic degradation of moxifloxacin by means of a stochastic cellular automaton, Chem. Eng. J. 188 (2012) 181–190.

[115] P. Van der Weeën, N. De Clercq, J. M. Baetens, C. Delbaere, K. Dewettinck, B. De Baets, A discrete stochastic model for oil migration in chocolate-coated confectionery, J. Food Eng. 117 (2013) 602–610.

[116] K. Scheerlinck, V. R. N. Pauwels, H. Vernieuwe, B. De Baets, Calibration of a water and energy balance model: Recursive parameter estimation versus particle swarm optimization, Water Resour. Res. 45 (2009) W10422.

[117] C. Y. T. Ma, D. K. Y. Yau, N. K. Yip, N. S. V. Rao, J. Chen, Stochastic steepest descent optimization of multiple-objective mobile sensor coverage, IEEE T. Veh. Technol. 61 (2012) 1810–1822.

[118] I. M. Sobol, Distribution of points in a cube and approximate evaluation of integrals, U.S.S.R Comput. Maths. Math. Phys. 7 (1967) 86–112.

[119] A. F. Martins, T. G. Vasconcelos, D. M. Henriques, C. S. Frank, A. König, K. Kümmerer, Concentration of ciprofloxacin in Brazilian hospital effluent and preliminary risk assessement: a case study, Clean-Soil Air Water 3 (2008) 264–269.

[120] M. Saquib, M. Muneer, TiO(2)-mediated photocatalytic degradation of triphenylmethane dye (gentian violet), in aqueous suspensions, Dyes Pigments 56 (2003) 37–49.

[121] H. Chun, W. Yizhong, T. Hongxiao, Destruction of phenol aqueous solution by photocatalysis or direct photolysis, Chemosphere 41 (2000) 1205–1209.

[122] R. W. Hartel, Chocolate: fat bloom during storage. The influence of structural elements, The Manufacturing Confectioner 79 (1999) 89–99.

[123] P. Lonchampt, R. W. Hartel, Fat bloom in chocolate and compound coatings, Eur. J. Lipid Sci. Technol. 106 (2004) 241–274.

[124] R. S. Khan, D. Rousseau, Hazelnut oil migration in dark chocolate - kinetic, thermodynamic and structural considerations, Eur. J. Lipid Sci. Technol. 108 (2006) 434–443.

[125] J. M. Aguilera, M. Michel, G. Mayor, Fat migration in chocolate: Diffusion or capillary flow in a particulate solid? A hypothesis paper, J. Food Sci. 69 (2004) 167–174.

[126] C. Loisel, G. Lecq, G. Ponchel, G. Keller, M. Ollivon, Fat bloom and chocolate structure studied by mercury porosimetry, J. Food Sci. 62 (1997) 781–788.

[127] T. Motwani, W. Hanselmann, R. C. Anantheswaran, Diffusion, counter-diffusion and lipid phase changes occuring during oil migration in model confectionery systems, J. Food Eng. 104 (2011) 186–195.

[128] J. Kleinert, Etude sur la formation du blanchiment gras et les moyens propres à le retarder, Rev. Int. Choc. 16 (1961) 345–368.

[129] N. D. Clercq, Changing the functionality of cocoa butter, Ph.D. thesis, Ghent University (2011).

[130] B. Wünderlich, Thermal Analysis, Academic Press, New York, 1990.

[131] H. Niederreiter, Low-discrepancy and low-dispersion sequences, J. Number Theory 30 (1998) 51–70.

[132] J. R. Galdámez, K. Szlachetka, J. L. Duda, G. R. Ziegler, Oil migration in chocolate: A case of non-Fickian diffusion, J. Food Eng. 92 (2009) 261–268.

[133] G. R. Ziegler, A. Shetty, R. C. Anantheswaran, Nut oil migration through chocolate, Manufacturing Confectioner 84 (2004) 118–126.

[134] D. Ma, M. Shao, Simulating infiltration into stony soils with a dual-porosity model, Eur. J. Soil Sci. 59 (2008) 950–959.

[135] J. R. Philip, The infiltration joining problem, Water Resour. Res. 23 (1987) 2239–2245.

[136] G. Ziegleder, I. Schwingshandl, Kinetics of fat migration within chocolate products. Part III: fat bloom, Fett-Lipid 100 (1998) 411–415.

[137] M. E. Miquel, S. Carli, P. J. Couzens, H.-J. Wille, L. D. Hall, Kinetics of the migration of lipids in composite chocolate measured by magnetic resonance imaging, Food Res. Int. 34 (2001) 773–781.

[138] L. Yu, D. Qi, Applying multifractal spectrum combined with fractal discrete Brownian motion model to wood defects recognition, Wood Sci. Technol. 45 (2011) 511–519.

[139] E. W. Washburn, The dynamics of capillary flow, Phys. Rev. 17 (1921) 273–283.

[140] P. Altamiras, L. Pyle, P. Bouchon, Structure-fat migration relationships during storage of cocoa butter model bars: Bloom development and possible mechanisms, J. Food Eng. 80 (2007) 600–610.

[141] J. R. Davis, Corrosion: Understanding the Basics, ASM International, 2000.

[142] P. R. Roberge, Corrosion Engineering: Principles and Practice, McGraw-Hill Professional, 2008.

[143] World Corrosion Organization - WCO.
URL `http://www.corrosion.org/`

[144] G. Contreras, S. Goidanich, S. Maggi, C. Piccardi, M. V. Diamanti, M. P. Pedeferri, L. Lazzari, Representing localized corrosion processes through cellular automata, Corros. Rev. 29 (2011) 241–245.

[145] R. Landolfo, L. Cascini, F. Portiolo, Modeling of metal structure corrosion damage: A state of the art report, Sustainability 2 (2010) 2163–2175.

[146] P. Marcus, V. Maurice, H. Strehblow, Localized corrosion (pitting): A model of passivity breakdown including the role of the oxide layer nanostructure, Corros. Sci. 50 (2008) 2698–2704.

[147] T. P. Hoar, The production and breakdown of the passivity of metals, Corros. Sci. 7 (1967) 341–355.

[148] S. M. Sharland, A review of the theoretical modelling of crevice and pitting corrosion, Corrosion 27 (1987) 289–323.

[149] G. S. Frankel, Pitting corrosion of metals: A review of the critical factors, J. Electrochem. Soc. 145 (1998) 2186–2198.

[150] Z. Szklarska-Smialowska, Pitting corrosion of metals, NACE, Houston, Texas, 1986.

[151] G. Riedel, C. Voigt, H. Werner, K. Erkel, M. Günzel, The influence of acid soluble sulfide inclusions on the passivation behaviour of austenitic Cr-Ni stainless steel, Corros. Sci. 27 (1987) 533–544.

[152] Z. Y. Liu, X. G. Li, Y. F. Cheng, In-situ characterization of the electrochemistry of grain and grain boundary of an X70 steel in a near-neutral pH solution, Electrochem. Commun. 12 (2010) 936–938.

[153] L. L. Shreir, R. A. Jarman, G. T. Burstein, Corrosion: Corrosion Control, Butterworth-Heinemann, 1994.

[154] I. A. Chaves, R. E. Melchers, Pitting corrosion in pipeline steel weld zones, Corros. Sci. 53 (2011) 4026–4032.

[155] M. Reuter, K. E. Heusler, Statistical investigations of the pitting of passive iron, Electrochim. Acta 35 (1990) 1809–1814.

[156] V. Vignal, H. Krawiec, O. Heintz, R. Oltra, The use of local electrochemical probes and surface analysis methods to study the electrochemical behaviour and pitting corrosion of stainless steels, Electrochim. Acta. 52 (2007) 4994–5001.

[157] S. M. Ghahari, A. J. Davenport, T. Rayment, T. Suter, J. Tinnes, C. Padovani, J. A. Hammons, M. Stampanoni, F. Marone, R. Mokso, In situ synchrotron X-ray micro-tomography study of pitting corrosion in stainless steel, Corros. Sci. 53 (2011) 2684–2687.

[158] L. J. Korb, Metals Handbook, Ninth edition, Asm Intl, 1987.

[159] B. Malki, B. Baroux, Computer simulation of the corrosion pit growth, Corros. Sci. 47 (2005) 171–182.

[160] H. W. Pickering, The significance of the local electrode potential within pits, crevices and cracks, Corros. Sci. 29 (1989) 325–341.

[161] H. Wang, B. Song, L. Wang, G. Lv, Three dimensional cellular automaton for simulation of the multi-pi corrosion damage evolution, Adv. Sci. Lett. 4 (2011) 2980–2985.

[162] A. Jarrah, M. Bigerelle, G. Guillemot, D. Najjar, A. Iost, J. Nianga, A generic statistical methodology to predict the maximum pit depth of a localized corrosion process, Corros. Sci. 53 (2011) 2453–2467.

[163] C. Punckt, M. Bölscher, H. H. Rotermund, A. S. Mikhailov, L. Organ, N. Budiansky, J. R. Scully, J. L. Hudson, Sudden onset of pitting corrosion on stainless steel as a critical phenomenon, Science 305 (2004) 1133–1136.

[164] A. M. Zimer, M. A. Carra, E. C. Rios, E. C. Pereira, L. H. Mascaro, Initial stages of corrosion pits on AISI 1040 steel in sulfide solution analyzed by temporal series micrographs coupled with electrochemical techniques, Corros. Sci. 76 (2013) 27–34.

[165] Y. González-García, G. T. Burstein, S. González, R. M. Souto, Imaging metastable pits on austenitic stainless steel in situ at the open-circuit corrosion potential, Electrochem. Commun. 6 (2004) 637–642.

[166] A. Turnbull, D. A. Horner, B. J. Connolly, Challenges in modelling the evolution of stress corrosion cracks from pits, Eng. Fract. Mech. 76 (2009) 633–640.

[167] D. E. Williams, C. Westcott, M. Fleischmann, Studies of the initiation of pitting corrosion on stainless steels, J. Electroanal. Chem. 180 (1984) 549–564.

[168] T. Okada, A theory of pertubation-initiated pitting, J. Electrochem. Soc. 125 (1978) 1382–1388.

[169] B. G. Ateya, H. W. Pickering, Effects of ionic migration on the concentrations and mass-transfer rate in the diffusion layer of dissolving metals, J. Appl. Electrochem. 11 (1981) 453–461.

[170] T. R. Beck, E. A. Grens, An electrochemical mass transport-kinetic model for stress corrosion cracking of titanium, J. Electrochem. Soc. 116 (1969) 177–184.

[171] S. M. Sharland, C. P. Jackson, A. J. Driver, A finite-element model of the propagation of corrosion crevices and pits, Corros. Sci. 29 (1989) 1149–1166.

[172] P. Smith, S. Roy, S. Swailes, S. Maxwell, D. Page, J. Lawson, A model for the corrosion of steel subjected to synthetic produced water containing sulfate, chloride and hydrogen sulfide, Chem. Eng. Sci. 66 (2011) 5775–5790.

[173] J. Cai, R. A. Cottis, S. B. Lyon, Phenomenological modelling of atmosperic corrosion using an artificial neural network, Corros. Sci. 41 (1999) 2001–2030.

[174] W. Zhang, S. Ruan, D. A. Wolfe, G. S. Frankel, Statistical model for intergranular corrosion growth kinetics, Corros. Sci. 45 (2003) 353–370.

[175] J. Stafiej, D. di Caprio, L. Bartosik, Corrosion-passivation processes in a cellular automata based simulation study, J. Supercomput. 65 (2013) 697–709.

[176] I. G. Main, J. R. Henderson, P. G. Meredith, P. R. Sammonds, Self-organized criticality and fluid-rock interactions in the brittle field, Pure Appl. Geophys. 142 (1994) 529–543.

[177] S. Gobron, N. Chiba, 3D surface cellular automata and their applications, J. Visual. Comput. Animat. 10 (1999) 143–158.

[178] L. Bartosik, D. di Caprio, J. Stafiej, Cellular automata approach to corrosion and passivity phenomena, Pure Appl. Chem. 85 (2013) 247–256.

[179] J. Saunier, A. Chaussé, J. Stafiej, J. P. Badiali, Simulations of diffusion limited corrosion at the metal-environment interface, J. Electroanal. Chem. 563 (2004) 239–247.

[180] R. M. Pidaparti, M. J. Palakal, L. Fang, Cellular automata approach to aircraft corrosion growth, Int. J. Artif. Intell. Tools 14 (2005) 361–369.

[181] K. C. Lan, Y. T. Chen, T. C. Hung, H. M. Tung, G. P. Yu, Simulation of the growth of oxide layer of stainless steels with chromium using cellular automaton model: Verification and parameter study, Comp. Mater. Sci. 77 (2013) 139–144.

[182] A. M. Zimer, E. C. Rios, P. D. Mendes, W. N. Gonçalves, O. M. Bruno, E. C. Pereira, L. H. Mascaro, Investigation of AISI 1040 steel corrosion in H_2S solution containing chloride ions by digital image processing coupled with electrochemical techniques, Corros. Sci. 53 (2011) 3193–3201.

[183] A. M. Zimer, E. C. Rios, L. H. Mascaro, E. C. Pereira, Temporal series micrographs coupled with polarization curves to study pit corrosion, Electrochem. Commum. 13 (2011) 1484–1487.

[184] A. M. Zimer, M. A. Carra, L. H. Mascaro, E. C. Pereira, Temporal series of micrographs coupled with electrochemical techniques to analyze pitting corrosion of AISI 1040 steel in carbonate and chloride solutions, Electrochem. Acta. 117 (2014) 344–350.

[185] P. Córdoba-Torres, R. P. Nogueira, L. de Miranda, L. Brenig, J. Wallenborn, V. Fairén, Cellular automaton simulation of a simple corrosion mechanism: mesoscopic heterogeneity versus macroscopic homogeneity, Electrochim. Acta 46 (2001) 2975–2989.

[186] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (2012) 281–305.

[187] K. Beven, J. Freer, Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the GLUE methodology, J. Hydrol. 249 (2001) 11–29.

[188] J. Van Impe, P. A. Vanrolleghem, D. Iserentant (Eds.), Advanced Instrumentation, Data Interpretation and Control of Biotechnological Processes, Kluwer, 1998.

[189] H. A. Gutowitz, A hierarchical classification of cellular automata, Physica D 45 (1990) 7–14.

[190] Š. Papáček, S. Čelikovský, B. Rehák, D. Štys, Experimental design for parameter estimation of two time-scale model of photosynthesis and photoinhibition in microalgae, Math. Comput. Simulat. 80 (2010) 1302–1309.

[191] W. Lin, F. Le Dimet, P. de Reffye, H. Bao-Gang, P. Cournegravede, K. Meng-Zhen, An optimal control methodology for plant growth - Case study of a water supply problem of sunflower, Math. Comput. Simulat. 2012 (2012) 909–923.

[192] M. F. Cardoso, R. L. Salcedo, S. F. de Azevedo, The simplex-simulated annealing approach to continuous non-linear optimization, Comput. Chem. Eng. 20 (1996) 1065–1080.

[193] V. C. Mariani, L. D. Coelho, A hybrid shuffled complex evolution approach with pattern search for unconstrained optimization, Math. Comput. Simulat. 81 (2011) 1901–1909.

[194] H. Levy, Stochastic Dominance: Investment Decision Making under Uncertainty, Springer, 2006.

[195] T. Hales, Dense Sphere Packings: A Blueprint for Formal Proofs, London Mathematical Society Lecture Note Series (Book 400), Cambridge University Press, 2012.

[196] P. F. O. Lindner, B. Hitzmann, Experimental design for optimal parameter estimation of an enzyme kinetic process based on the analysis of the Fisher information matrix, J. Theor. Biol. 238 (2006) 111–123.

[197] D. J. Daley, J. Gani, Epidemic Modeling: An Introduction, Cambridge University Press, NY, 2005.

[198] Web of Science.
URL `http://apps.webofknowledge.com`

[199] L. B. Kier, C.-K. Cheng, Effect of initial temperature on water aggregation at a cold surface, Chem. Biodivers. 10 (2013) 138–143.

[200] P. Setny, M. Zacharias, Hydration in discrete water. A mean field, cellular automata based approach to calculating hydration free energies, J. Phys. Chem. B 114 (2010) 8667–8675.

[201] C.-K. Cheng, L. B. Kier, Cellular-automata models of solid liquid interfaces, Chem. Biodivers. 4 (2007) 2555–2563.

[202] M. A. Pasquale, G. P. Saracco, S. L. Marchiano, A. J. Arvia, Influence of pinning effects on the electrochemical formation of silver patterns in agarose-containing sols and gels, J. Phys. Chem. B 109 (2005) 20256–20265.

[203] L. B. Kier, C.-K. Cheng, B. Testa, Studies of ligand diffusion pathways over a protein surface, J. Chem. Inf. Comput. Sci. 43 (2003) 255–258.

[204] L. B. Kier, C.-K. Cheng, B. Testa, A cellular automata model of ligand passage over a protein hydrodynamic landscape, J. Theor. Biol. 215 (2002) 415–426.

[205] A. Taleb, A. Chaussé, M. Dymitrowska, J. Stafiej, J. P. Badiali, Simulations of corrosion and passivation phenomena: Diffusion feedback on the corrosion rate, J. Phys. Chem. B 108 (2004) 952–958.

[206] A. Taleb, J. Stafiej, J. P. Badiali, Numerical simulation of crystallographic corrosion: Particle production and surface roughness, J. Phys. Chem. C 111 (2007) 9086–9094.

[207] L. Lei, L. Xiaogang, D. Chaofang, X. Kui, L. Lin, Cellular automata modeling on pitting current transients, Electrochem. Commum. 11 (2009) 1826–1829.

[208] A. Taleb, J. Stafiej, Numerical simulation of the effect of grain size on corrosion processes: Surface roughness oscillation and cluster detachment, Corros. Sci. 53 (2011) 2508–2513.

[209] X.-S. Yang, Computational modelling of nonlinear calcium waves, Appl. Math. Model. 30 (2010) 200–208.

[210] R. Toth, C. Stone, A. Adamatzky, B. de Lacy Costello, L. Bulle, Dynamic control and information processing in the Belousov-Zhabotinsky reaction using a coevolutionary algorithm, J. Chem. Phys. 129 (2008) 184708.

[211] H. Hallberg, M. Wallin, M. Ristinmaa, Simulation of discontinuous dynamic recrystallization in pure Cu using a probabilistic cellular automaton, Comp. Mater. Sci. 49 (2010) 25–34.

[212] T. J. Hutton, Evolvable self-replicating molecules in an artificial chemistry, Artif. Life 8 (2002) 341–356.

[213] S. Cao, K. L. Yeung, P. Yue, An investigation of trichloroethylene photocatalytic oxidation on mesoporous titania-silica aerogel catalysts, Appl. Catal., B. 76 (2007) 64–72.

[214] X. Yu, S. Chen, L. Wang, Simulation of recrystallization in cold worked stainless steel and its effect on chromium depletion by cellular automaton, Comp. Mater. Sci. 46 (2009) 66–72.

[215] S. Hayashi, Cellular automata for electrochemistry. Sweep rate, Electrochem. 81 (2013) 688–690.

[216] X. Xiao, K.-C. Chou, Using pseudo amino acid composition to predict protein attributes via cellular automata and other approaches, Curr. Bioinform. 6 (2011) 251–260.

[217] X. Xiao, P. Wang, K.-C. Chou, Predicting protein structural classes with pseudo amino acid composition: An approach using geometric moments of cellular automaton image, J. Theor. Biol. 254 (2008) 691–696.

[218] R. Yu, H. Chen, T. Chen, X. Zhou, Modeling and simulation of drug release from multi-layer biodegradable polymer microstructure in three dimensions, Simul. Model. Pract. Th. 16 (2008) 15–25.

[219] S. M. Hunt, M. A. Hamilton, J. T. Sears, G. Harkin, J. Reno, A computer investigation of chemically mediated detachment in bacterial biofilms, Microbiology 149 (2003) 1155–1163.

[220] P. Van der Weeën, A. M. Zimer, E. C. Pereira, L. H. Mascaro, O. M. Bruno, B. De Baets, Modeling pitting corrosion by means of a 3D discrete stochastic model, Corros. Sci. 82 (2014) 133–144.

[221] D. S. Svyetlichnyy, Modeling of grain refinement by cellular automata, Comp. Mater. Sci. 77 (2013) 408–416.

[222] L. Mao, H. H. Harris, K. J. Stine, Simple lattice simulation of chiral discrimination in monolayers, J. Chem. Inf. Comput. Sci. 42 (2002) 1179–1184.

[223] D. DeSoi, L. B. Kier, C.-K. Cheng, H. T. Karnes, An expanded cellular automata model for enantiomer separations using a β-cyclodextrin stationary phase, J. Chromatogr. A 1291 (2013) 73–83.

[224] V. Alopaeus, K. Hynynena, J. Aittamaa, A cellular automata model for liquid distribution in trickle bed reactors, Chem. Eng. Sci. 61 (2006) 4930–4943.

[225] H. Z. Sohi, B. Khoshandam, Cellular automata modeling of noncatalytic gas-solid reactions, Chem. Eng. J. 202 (2012) 710–719.

[226] B. de Lacy Costello, R. Toth, Implementation of glider guns in the light-sensitive Belousov-Zhabotinsky medium, Phys. Rev. E 79 (2009) 026114.

[227] F. Jiménez-Morales, M. C. Lemos, A cellular automaton for the catalyzed oxidation of carbon monoxide: the effect of the temperature, Phys. Scripta 118 (2005) 174–178.

[228] Q. Liao, Y.-J. Wang, Y.-Z. Wang, R. Chen, X. Zhu, Y.-K. Pu, D.-J. Lee, Two-dimension mathematical modeling of photosynthetic bacterial biofilm growth and formation, Int. J. Hydrogen Energ. 37 (2012) 15607–15615.

[229] J. R. Weimar, Three-dimensional cellular automata for reaction-diffusion systems, Fund. Inform. 52 (2002) 275–282.

[230] T. J. Laaksonen, H. M. Laaksonen, J. T. Hirvonen, L. Murtomäki, Cellular automata model for drug release from binary matrix and reservoir polymeric devices, Biomaterials 30 (2000) 1978–1987.

[231] A. C. J. de Korte, H. J. H. Brouwers, A cellular automata approach to chemical reactions; 1 Reaction controlled systems, Chem. Eng. J. 228 (2013) 172–178.

[232] H. Hallberg, M. Ristinmaa, Microstructure evolution influenced by dislocation density gradients modeled in a reaction-diffusion system, Comp. Mater. Sci. 67 (2013) 373–383.

[233] J. W. Bullard, R. J. Flatt, New insights into the effect of calcium hydroxide precipitation on the kinetics of tricalcium silicate hydration, J. Am. Ceram. Soc. 93 (2010) 1894–1903.

[234] P. Córdoba-Torres, R. P. Nogueira, V. Fairén, Fractional reaction order kinetics in electrochemical systems involving single-reactant, bimolecular desorption reactions, J. Electroanal. Chem. 560 (2003) 25–33.

[235] K. Laurio, F. Linåker, A. Narayanan, Regular biosequence pattern matching with cellular automata, Inform. Sciences 46 (2002) 89–101.

[236] A. V. Dimaki, E. V. Shil'ko, S. G. Psakh'e, Simulating the propagation of exothermic reactions in heterogeneous media, Combust. Explo. Shock 41 (2005) 151–157.

[237] A. Nishiyama, T. Tokihiro, Construction of an isotropic cellular automaton for a reaction-diffusion equation by means of a random walk, J. Phys. Soc. Jpn. 80 (2010) 054003.

[238] Z. Zhou, Q. Hang, Z. Zhu, W. Li, An efficient simulation system for inclined UV lithography processes of thick SU-8 photoresists, IEEE T. Manufact. M. 24 (2011) 294–303.

[239] A. E. Bresciania, C. F. X. Mendonc, R. M. B. Alvesa, C. A. O. Nascimento, Modeling the kinetics of the coalescence of water droplets in crude oil emulsions subject to an electric field, with the cellular automata technique, Comput. Chem. Eng. 34 (2010) 1962–1968.

[240] L. B. Kier, R. M. Tombes, Proton hopping: A proposed mechanism for myelinated axon nerve impulses, Chem. Biodivers. 10 (2013) 596–599.

[241] O. Bandman, Using cellular automata for porous media simulation, J. Supercomput. 57 (2011) 121–131.

[242] D. Levy, T. Requeijo, Stochastic models for phototaxis, B. Math. Biol. 70 (2008) 1684–1706.

[243] G. J. Martínez, A. Adamatzky, B. de Lacy Costello, On logical gates in precipitating medium: Cellular automaton model, Phys. Lett. A 372 (2008) 5115–5119.

[244] D. di Caprio, J. Stafiej, The role of adsorption in passivation phenomena modelled by discrete lattice gas automata, Electrochim. Acta 56 (2011) 396–3968.

[245] A. Adamatzky, B. de Lacy Costello, Collision-free path planning in the Belousov-Zhabotinsky medium assisted by a cellular automaton, Naturwissenschaften 89 (2002) 474–478.

[246] N. Bertrand, G. Leclair, P. Hildgen, Modeling drug release from bioerodible microspheres using a cellular automaton, Int. J. Pharm. 343 (2007) 196–207.

[247] H. Beyenal, Z. Lewandowski, Modeling mass transport and microbial activity in stratified biofilms, Chem. Eng. Sci. 60 (2005) 4337–4348.

[248] S.-A. Marashi, M. Kargar, A. Katanforoush, H. Abolhassani, M. Sadeghib, Evolution of ligand-diffusion chreodes' on protein-surface models: A genetic-algorithm study, Chem. Biodivers. 4 (2007) 2766–2771.

[249] L. Bi, J. Kim, L. Wen, A. Kumar, M. Fulham, D. D. Feng, Cellular automata and anisotropic diffusion filter based interactive tumor segmentation for positron emission tomography, in: Conf. Proc. IEEE Eng. Med. Biol. Soc., 2013.

[250] A. Alexandridis, D. Vakalis, C. I. Siettos, G. V. Bafas, A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990, Appl. Math. Comput. 204 (2008) 191–201.

[251] B. Testa, L. B. Kier, C.-K. Cheng, A cellular automata model of water structuring by a chiral solute, J. Chem. Inf. Comput. Sci. 42 (2002) 712–716.

[252] L. B. Kier, D. Bonchev, G. A. Buck, Modeling biochemical networks: A cellular-automata approach, Chem. Biodivers. 2 (2005) 233–243.

[253] V. Y. Shevchenko, S. V. Krivovichev, A. L. Mackay, Cellular automata and local order in the structural chemistry of the lovozerite group minerals, Glass Phys. Chem. 36 (2010) 1–9.

[254] S. Hayashi, Cellular automata for electrochemistry. Models of fast catalytic reactions, Electrochem. 81 (2013) 961–965.

[255] Y. Diao, D. Ma, Z. Wen, J. Yin, J. Xiang, M. Li, Using pseudo amino acid composition to predict transmembrane regions in protein: cellular automata and Lempel-Ziv complexity, Amino Acids 34 (2008) 111–117.

[256] J. W. Bullard, A determination of hydration mechanisms for tricalcium silicate using a kinetic cellular automaton model, J. Am. Ceram. Soc. 91 (2008) 2088–2097.

[257] D. DeSoi, L. B. Kier, C.-K. Cheng, H. T. Karnes, A cellular automata model of enantiomer binding strengths to β-cyclodextrin, J. Chromatogr. A 1258 (2012) 101–107.

[258] H. Chen, Y. Chen, J. Zhang, Cellular automaton modeling on the corrosion/oxidation mechanism of steel in liquid metal environment, Prog. Nucl. Energ. 50 (2008) 587–593.

[259] G. Chao, S. Xiaobo, C. Chenglin, D. Yinsheng, P. Yuepu, L. Pinghua, A cellular automaton simulation of the degradation of porous polylactide scaffold: I. Effect of porosity, Mater. Sci. Eng. 29 (2009) 1950–1958.

[260] N. Fatès, Solving the decentralised gathering problem with a reaction-diffusion-chemotaxis scheme, Swarm Intell. 4 (2010) 91–115.

[261] H. Wang, H. Zhao, Z. Guo, Y. He, C. Zheng, Lattice boltzmann method for simulations of gas-particle flows over a backward-facing step, J. Comput. Phys. 239 (2013) 57–71.

[262] M. Fathi, M. Mohebbi, J. Varshosaz, F. Shahidi, Cellular automata modeling of hesperetin release phenomenon from lipid nanocarriers, Food Bioprocess. Technol. 6 (2013) 3134–3142.

[263] C. Sarkar, S. A. Abbasi, Enhancing the accuracy of forecasting impact of accidents in chemical process industry by the application of cellular automata technique, Process Saf. Environ. 84 (2006) 355–370.

[264] R. J. Doran, S. W. Laffan, Simulating the spatial dynamics of foot and mouth disease outbreaks in feral pigs and livestock in Queensland, Australia, using a susceptible-infected-recovered cellular automata model, Prev. Vet. Med. 70 (2005) 133–157.

[265] G. C. Sirakoulis, I. Karafyllidis, A. Thanailakis, A cellular automaton model for the effects of population movement and vaccination on epidemic propagation, Ecol. Modell. 133 (2000) 209–223.

[266] C. Vautrin-Ul, A. Taleb, J. Stafiej, A. Chaussé, J. P. Badiali, Reprint of 'Mesoscopic modelling of corrosion phenomena: Coupling between electrochemical and mechanical processes, analysis of the deviation from the Faraday law', Electrochim. Acta 52 (2007) 780–7810.

[267] Y. Zhao, S. A. Billings, D. Coca, R. I. Ristic, L. DeMatos, Identification of the transition rule in a modified cellular automata model: The case of dendritic NH_4Br crystal growth, Int. J. Bifurcat. Chaos 19 (2009) 2295–2305.

[268] Y. Zhao, R. S. Qin, D. F. Chen, A three-dimensional cellular automata model coupled with finite element method and thermodynamic database for alloy solidification, J. Cryst. Growth 377 (2013) 72–77.

[269] A. Adamatzky, B. de Lacy Costello, Reaction-diffusion path planning in a hybrid chemical and cellular-automaton processor, Chaos Soliton Fract. 16 (2003) 727–736.

[270] K. C. Iarosz, A. M. Batista, R. L. Viana, S. R. Lopes, I. L. Caldas, T. J. P. Penna, The influence of connectivity on the firing rate in a neuronal network with electrical and chemical synapses, Physica A 391 (2012) 819–827.

[271] L. B. Kier, C.-K. Cheng, J. D. Nelson, Cellular automata models of the influences on solute diffusion through water, Chem. Biodivers. 6 (2009) 295–302.

[272] P. Van der Weeën, J. M. Baetens, B. De Baets, Design and parameterization of a stochastic cellular automaton describing a chemical reaction, J. Comput. Chem. 32 (2011) 1952–1961.

[273] L. B. Kier, L. H. Hall, Molecular connectivity: intermolecular accessibility and encounter simulation, J. Mol. Graph. Model. 20 (2001) 76–83.

[274] J. X. Lin, C. Y. Wang, Y. Y. Zheng, Prediction of isothermal crystallization parameters in monomer cast nylon 6, Comp. Chem. Eng. 32 (2008) 3023–3029.

[275] D. Siegismund, T. F. Keller, K. D. Jandt, M. Rettenmayr, Fibrinogen adsorption on biomaterials - A numerical study, Macromol. Biosci. 10 (2010) 1216–1223.

[276] T. Czárána, R. F. Hoekstra, A spatial model of the evolution of quorum sensing regulating bacteriocin production, Behav. Ecol. 18 (2007) 866–873.

[277] X. Yu, S. Chen, Y. Liu, F. Ren, A study of intergranular corrosion of austenitic stainless steel by electrochemical potentiodynamic reactivation, electron back-scattering diffraction and cellular automaton, Corros. Sci. 52 (2010) 1939–1947.

[278] C. Echeverria, W. Olivares-Rivas, K. Tucci, Cellular automata simulation of the spatial conformations of polyelectrolytes, Ciencia 14 (2006) 452–458.

[279] R. L. Viana, F. S. Borges, K. C. Iarosz, A. M. Batista, S. R. Lopes, I. L. Caldas, Dynamic range in a neuron network with electrical and chemical synapses, Commun. Nonlinear Sci. Numer. Simulat. 19 (2013) 164–172.

[280] A. Yazdan, G. Mehrdad, M. M. Ghasem, Cellular automata simulation of a bistable reaction-diffusion system: Microscopic and macroscopic approaches, Iran. J. Chem. Chem. Eng. 30 (2011) 143–150.

[281] A. Neuforth, P. G. Seybold, L. B. Kier, C.-K. Cheng, Cellular automata models of kinetically and thermodynamically controlled reactions, Int. J. Chem. Kinet. 32 (2000) 529–534.

[282] K. Odagiri, K. Takatsuka, Threshold effect with stochastic fluctuation in bacteria-colony-like proliferation dynamics as analyzed through a comparative study of reaction-diffusion equations and cellular automata, Phys. Rev. E 79 (2009) 026202.

[283] A. C. Warden, B. A. Little, V. S. Haritos, A cellular automaton model of crystalline cellulose hydrolysis by cellulases, Biotechnol. Biofuels 4 (2011) 39–53.

[284] L. B. Kier, C.-K. Cheng, T. Karnes, A cellular automata model of chromatography, Biomed. Chromatogr. 14 (2000) 530–534.

[285] P. Córdoba-Torres, R. P. Nogueira, V. Fairén, Forecasting interface roughness from kinetic parameters of corrosion mechanisms, J. Electroanal. Chem. 529 (2002) 109–123.

[286] I. Bena, M. Droz, K. Martens, Z. Rácz1, Reaction-diffusion fronts with inhomogeneous initial conditions, J. Phys.: Condens. Matter 19 (2007) 065103.

[287] S. Branciamore, E. Gallori, E. Szathmáry, T. Czárán, The origin of life: Chemical evolution of a metabolic system in a mineral honeycomb?, J. Mol. Evol. 69 (2009) 458–469.

[288] P. Dini, C. L. Nehaniv, A. Egri-Nagy, M. J. Schilstra, Exploring the concept of interaction computing through the discrete algebraic analysis of the Belousov-Zhabotinsky reaction, Biosystems 112 (2013) 145–162.

[289] A. Adamatzky, B. de Lacy Costello, C. Melhuish, N. Ratcliffe, Experimental reaction-diffusion chemical processors for robot path planning, J. Intell. Robot. Syst. 37 (2003) 233–249.

[290] S. V. Krivovichev, Algorithmic crystal chemistry: A cellular automata approach, Crystallogr. Rep. 57 (2012) 10–17.

Curriculum Vitae

Personal information

Name: Pieter Van der Weeën
Nationality: Belgian
Gender: Male
Date of birth: 07/05/1987
Place of birth: Aalst, Belgium
Telephone: 0032 494 22 38 64
Email: pieter.vanderweeen@gmail.com

Experience

August 2014 - Present

Company: Oleon nv
Function: Junior R& D project leader food and feed

Education and research

October 2010 - September 2014

Institution: Ghent University
Degree: Doctor of Applied Biological Sciences
Doctoral thesis: Meso-scale modeling of reaction-diffusion processes using cellular automata
Funding: BOF scholarship (Bijzonder Onderzoeksfonds UGent)

Scientific output

- P. Van der Weeën and B. De Baets (2014). Cellular automata in chemical engineering: critical evaluation and good practice guidelines, Chemical Engineering Science, Revision submitted.

- P. Van der Weeën, J. Verwaeren and B. De Baets (2014). A search grid for parameter optimization as a byproduct of model sensitivity analysis, Applied Mathematics and Computation, Submitted.
- N. R. da Silva, P. Van der Weeën, B. De Baets and O. M. Bruno (2014). Improved texture image classification through the use of a corrosion-inspired cellular automaton, Neurocomputing, Accepted.
- P. Van der Weeën, A. M. Zimer, E. C. Pereira, O. M. Bruno, L. H. Mascaro and B. De Baets (2014). Modeling pitting corrosion by means of a 3D discrete stochastic model, Corrosion Science, 82, 133-144.
- P. Van der Weeën, N. De Clercq, J. M. Baetens, C. Delbaere, K. Dewettinck and B. De Baets (2013). A discrete stochastic model for oil migration in chocolate-coated confectionery, Journal of Food Engineering, 119, 602-610.
- P. Van der Weeën, J. M. Baetens, J. Verwaeren, X. Van Doorslaer, P. M. Heynderickx, J. Dewulf and B. De Baets (2012). Modeling the photocatalytic degradation of moxifloxacin by means of a stochastic cellular automaton, Chemical Engineering Journal, 188, 181-190.
- J. M. Baetens, P. Van der Weeën and B. De Baets (2012). Effect of asynchronous updating on the stability of cellular automata, Chaos, Solitons & Fractals, 45, 383-394.
- P. Van der Weeën, J. M. Baetens and B. De Baets (2011). Design and parameterization of a stochastic cellular automaton describing a chemical reaction, Journal of Computational Chemistry, 32, 1952-1961.

Oral presentations

- P. Van der Weeën and B. De Baets, Influence of the grid resolution on output accuracy and parameter sensitivity, Summer Solstice 2014, International Conference on Discrete Models of Complex Systems, Ljubljana (Slovenia), June 24th 2014.
- P. Van der Weeën, J. Verwaeren and B. De Baets, Designing a search grid for parameter estimation using sensitivity analysis, AMMCS 2013, Conference on Applied Mathematics, Modeling, and Computer Science, Waterloo (Canada), August 27th 2013.
- P. Van der Weeën, J. M. Baetens and B. De Baets, A discrete stochastic model for pitting corrosion, AMMCS 2013, Conference on Applied Mathematics, Modeling, and Computer Science, Waterloo (Canada), August 30th 2013.
- P. Van der Weeën, N. De Clercq, K. Dewettinck and B. De Baets, A CA-based model describing fat bloom in chocolate, ACRI 2012, Cellular Automata for Researsch and Industry, Santorini (Greece), September 26th 2012.
- P. Van der Weeën, J. M. Baetens, X. Van Doorslaer, P. M. Heynderickx, J. Dewulf and B. De Baets, A cellular automata approach for modeling the photocatalytic degradation of fluoroquinolones, Summer Solstice 2011, International Conference on Discrete Models of Complex Systems, Turku (Finland), June 27th, 2011.

- P. Van der Weeën, J. M. Baetens and B. De Baets, Parameterization of a stochastic cellular automaton describing a chemical reaction, NSABS 2010, National Symposium on Applied Biological Sciences, Ghent (Belgium), December 20th 2010.

Achievements

- Second prize for best student presentation: Applied Mathematics, Modeling and Computational Science (AMMCS), 2013, Waterloo, Canada.
- Procter & Gamble Modeling and Simulation award: 16th Symposium on Applied Biological Sciences, 2010, Ghent, Belgium.
- Award for best oral presentation: 16th Symposium on Applied Biological Sciences, 2010, Ghent, Belgium.

Others

I constantly strive to learn new things. During my doctoral training I broadened my knowledge on:

- Economics: Micro and Macro Economy (UGent course, 2011), Marketing Management (UGent course, 2011), Accountancy (UGent course, 2012), Technology and innovation management (UGent course, 2013) and Managerial skills (UGent course, 2014).
- Pharmaceutical Bioinformatics (course at Swedish University of Uppsala, 2013).
- Project Management (doctoral school course, 2011).
- Sensitivity Analysis (summer school of the Joint Research Centre of European Commission, 2012).

September 2008 - June 2010

Institution:	Ghent University
Degree:	Master in Bioscience Engineering, option: Chemistry and biotechnology (Great Distinction)
Master thesis:	Development and validation of discrete spatial-temporal models for the description of chemical reactions
Internship:	Procter & Gamble (Strombeek-Bever, Belgium): Influence of process parameters on deposition kinetics in fabric softeners.

September 2005 - June 2008

Institution:	Ghent University
Degree:	Bachelor in Bioscience Engineering, option: Chemistry and food science (Great Distinction)
Bachelor thesis:	Engineers searching for inspiration in nature

Skills

Languages

Native:	Dutch
Advanced:	English, Spanish
Good:	French
Basic:	German

Computer

Software:	Microsoft Office, Mathematica, Matlab & Simulink, R, Maple, Latex & Beamer, Inkscape, Bioclipse, SPSS and Python
Programming:	Basics of Java, UNIX shell scripts, HTML and Linux

Competences

Key competences include: Analysis, communication, organization and management

Miscellaneous

On the board of Verkeerd Geparkeerd VZW for five years (2007-2012), of which two years as chair (2010-2012)